



ERROR SOLUTIONS

REF. 0402

CNC8070



目录

ERRORS 0000-0999 1
ERRORS 1000-1999 9
ERRORS 2000-2999 79
ERRORS 3000-3999 85
ERRORS 4000-4999 93
ERRORS 5000-5999 97
ERRORS 6000-6999 99
ERRORS 7000-7999 105
ERRORS 8000-8999 111
TOOL AND TOOL MAGAZINE TABLE 121
MESSAGES OF THE PROFILE EDITOR 125



0000-0999 号错误

0001 ' 系统错误 '

检测时间:	执行程序时。
引起原因:	软件或硬件错误导致数据损坏和 (或) 结果不一致。
解决方案:	此类错误通常会强制 CNC 输出。 请与厂商联系。

0002 ' 系统警告 '

检测时间:	执行程序时。
引起原因:	对可能演化成系统错误的内部状况报警。
解决方案:	通常可通过 "ESCAPE" 键恢复正常。 请与厂商联系。

0003 ' 请求内存错误 '

检测时间:	CNC 上电时。
引起原因:	CNC 内存不够。
解决方案:	请与厂商联系。

0004 'PLC 数据校验和错误 '

检测时间:	CNC 上电时。
引起原因:	存入盘中的 PLC 数据无效。有可能是包含数据 (\FagorCNC\ PLC8070\ User\ project\ objects\ plcData.bin) 的文件丢失而不能访问，也有可能文件已经损坏。这样会导致某些 PLC 数据 (标记、计数器、定时器和寄存器) 丢失。
解决方案:	如果重新启动 CNC 数次后仍然不能排除该错误，请与厂商联系。

0005 'CNC 非法关闭，必须回零 '

检测时间:	CNC 上电时。
引起原因:	存入盘中的 CNC 数据无效 (坐标、偏置、PARTC)。有可能是包含数据 (\FagorCNC\ Data\ orgData.tab) 的文件丢失而导致不能访问，也有可能文件已经损坏。这样会导致某些数据丢失。它会丢掉坐标、偏置、PARTC 和动力学等值。
解决方案:	如果重新启动 CNC 数次后仍然不能排除该错误，请与厂商联系。

0020 ' 访问变量错误 '

检测时间:	CNC 上电或界面换页时。
引起原因:	CNC 不能识别所访问的变量。
解决方案:	联系机床制造商或界面屏幕设计人员，删除对 CNC 不能识别的变量的访问。



0040 '程序段中 M 功能前－前或前－后有子程序的不允许编写运动指令'

检测时间： 机床参数确认。

引起原因： 具有关联子程序的 M 功能不能为前－前或前－后。与 M 功能关联的子程序始终要在程序段的末尾执行，即在程序段其他部分后面。

- 前－前表示M功能在程序段其他部分前执行，与PLC的同步同样也在程序段前完成。
- 前－后表示 M 功能在程序段其他部分前执行，而与 PLC 的同步则在程序段后完成。

解决方案： 定义不带子程序的 M 功能或定义 M 功能为后－后。

0041 '表格中 M 功能重复'

检测时间： 机床参数确认。

引起原因： M 功能表格中有重复 M 功能。

解决方案： 从表格中删除重复的 M 功能。

0042 '机床参数值错误'

检测时间： 机床参数确认。

引起原因： 机床参数超出规定范围。

解决方案： 设置介于警告中提示的最大值和最小值之间的参数值。

0043 '重启 CNC 使新值生效'

检测时间： 机床参数确认。

引起原因： 机床参数被修改，需要重启 CNC 使新值生效。

解决方案： 重启 CNC。

0044 '轴名错误或未定义轴名'

检测时间： 机床参数确认。

引起原因： 将代表轴名的参数错误的赋给了某个机床参数。

解决方案： 为参数赋予正确的名称。第一个字符必须为 A、 B、 C、X、 Y、 Z、 U、 V 或 W，第二个字符可以为 1 ～ 9 之间的数字。

0045 '主轴名错误'

检测时间： 机床参数确认。

引起原因： 将代表主轴名的参数错误的赋给了某个机床参数。

解决方案： 为参数赋予正确的名称。第一个字符必须为S，第二个字符可以为1～9之间的数字。

0046 '轴不存在'

检测时间： 机床参数确认。

引起原因： 将代表轴名的机床参数赋给了 AXISNAME 中不存在的某个名称。

解决方案： 为参数赋予正确的名称。名称必须存在于 AXISNAME 中。

0047 '不能将主轴定义为从轴'

检测时间： 机床参数确认。

引起原因： 将另一龙门对中的主轴设为本龙门对中的从轴。

解决方案： 不要将其设置为从轴。

0048 '一根轴不能同时为几个主轴的从轴'

检测时间： 机床参数确认。

引起原因： 将另一龙门对中的从轴设为本龙门对中的从轴。

解决方案： 不要重复设置其为从轴。



0049 " 主轴不能为从轴，反之亦然 '

检测时间： 机床参数确认。

引起原因： 某个龙门对中的从轴，不能成为另一龙门对中的主轴。同样的，某个龙门对中的主轴，不能成为另一龙门对中的从轴。

解决方案： 核对龙门轴表。

0050 ' 主轴和从轴必须为同一类型 (AXISTYPE)'

检测时间： 机床参数确认。

引起原因： 龙门对的轴必须拥有相同的机床参数 "AXISTYPE " (直线、旋转)。

解决方案： 核对龙门对中的龙门轴表或 "AXISTYPE" 参数。

0051 ' 主轴与从轴的某些参数值必须相同 '

检测时间： 机床参数确认。

引起原因： 如果为直线轴，那么必须具有相同值的机床参数包括：AXISMODE(轴模式)、FACEAXIS(曲面轴) 和 LONGAXIS(长轴)。如果为旋转轴，必须具有相同值的机床参数包括：AXISMODE(轴模式)、 SHORTESTWAY 和 CAXIS。

解决方案： 核对上面涉及的各个参数。

0052 ' 模块差太小 '

检测时间： 机床参数确认。

引起原因： 机床参数 MODUPLIM 和 MODLOWLIM 的差小于轴的分辨率。

解决方案： 增加 MODUPLIM 或减小 MODLOWLIM 的值。

0053 ' 几个手轮中的 MGPAXIS 参数重复 '

检测时间： 机床参数确认。

引起原因： 手动操作表中两个或两个以上的手轮拥有相同的轴名。

解决方案： 为每个手轮分配不同的轴名。

0054 'MOVAXIS 与 COMPAXIS 轴不能相同 '

检测时间： 机床参数确认。

引起原因： 在交叉补偿表格中，被补偿轴与补偿轴的轴相同。

解决方案： 分配不同的轴名。

0055 ' 轴导致交叉补偿错误并受其影响 '

检测时间： 机床参数确认。

引起原因： 核对交叉补偿表中被补偿 (受影响) 轴 (COMPAXIS) 与其运动影响其他轴的轴之间的关系，定义了一根轴受自己运动的影响，这是没有任何意义的。

解决方案： 在交叉补偿表中正确定义 MOVAXIS 和 COMPAXIS。

0056 ' 补偿表的位置没有按升序进行排列 '

检测时间： 机床参数确认。

引起原因： 补偿表中的位置 (POSITION) 参数值必须为升序。

解决方案： 在前一值和后一值之间赋值。

0057 ' 补偿表中误差斜率大于 1'

检测时间： 机床参数确认 (螺线误差补偿表)。

引起原因： 螺线补偿表中的误差斜率不能大于 1。换句话说，连续两个误差的差不能大于它们的间距。

解决方案： 根据最大间距编写误差。如果不这样，输入的螺线误差将会过大而不能进行补偿。

0058 ' 必须重启 CNC，以使 HMI 表中所作的修改生效 '

检测时间： 机床参数确认。

引起原因： 为使 HMI 表中所作的修改生效，必须重启 CNC。

解决方案： 重启 CNC。

0059 ' 必须重启 CNC，以使刀库表中所作修改生效 '

检测时间： 机床参数确认。

引起原因： 为使刀库表中所作修改生效，必须重启 CNC。

解决方案： 重启 CNC。

0060 '最大微动进给率大于轴的最大进给率'

检测时间： 机床参数确认。
引起原因： 机床参数 MAXMANFEED 大于 G00FEED。
解决方案： 减小 MAXMANFEED。

0061 '手动快速进给率大于轴的最大进给率'

检测时间： 机床参数确认。
引起原因： 机床参数 JOGRAPFEED 大于 G00FEED。
解决方案： 减小 JOGRAPFEED。

0062 '连续微动进给率大于轴的最大进给率'

检测时间： 机床参数确认。
引起原因： 机床参数 JOGFEED 大于 G00FEED。
解决方案： 减小 JOGFEED。

0063 '增量微动进给率大于轴的最大进给率'

检测时间： 机床参数确认。
引起原因： 机床参数 INCJOGFEED 大于 G00FEED。
解决方案： 减小 INCJOGFEED。

0064 '主轴和从轴必须具有相同的 I0TYPE'

检测时间： 机床参数确认。
引起原因： 龙门轴的 IO 类型不一致。
解决方案： 改变参数 I0TYPE 的值。

0065 'Hirth 轴不能为龙门轴 "

检测时间： 机床参数确认。
引起原因： CNC 不允许龙门轴对为 Hirth 轴。
解决方案： 移除龙门对或利用非 Hirth 轴构建龙门对。

0066 '龙门轴不能拥有 REFSHIFT'

检测时间： 机床参数确认。
引起原因： 龙门对的某根轴的机床参数 REFSHIFT(相对于参考点的偏置) 在其中一个设置 (参数设置) 中不为 0。
解决方案： REFSHIFT = 0。

0067 '龙门轴不能为单向'

检测时间： 机床参数确认。
引起原因： 只能单向旋转 (UNIDIR) 的轴不能为龙门轴。
解决方案： UNIDIR = 0。

0068 '龙门轴：AXISNAME 中从轴不能居于主轴前'

检测时间： 机床参数确认。
引起原因： 轴名分配表 (AXISNAME) 中从轴位于主轴之上。
解决方案： 在 AXISNAME 中交换轴名或交换龙门对中的主轴和从轴。

0069 '龙门轴：如果主轴没有 DECINPUT（回零开关），那么从轴也不能有'

检测时间： 机床参数确认。
引起原因： 龙门对的轴不能利用回零开关回零 (DECINPUT = TRUE)。
解决方案： DECINPUT = FALSE。

0070 ' 龙门轴 : 主轴和从轴的 LIMIT+ 和 LIMIT- 必须一致。 '

检测时间： 机床参数确认。
引起原因： 龙门轴的软件极限不一致。
解决方案： 为从轴设置与主轴一样的极限 (LIMIT+ 和 LIMIT-)。

0071 'CNC 未激活跟随误差监测 '

检测时间： 机床参数确认。
引起原因： 该监测未被激活。
解决方案： 设置轴参数 FLWEMONITOR = TRUE。

0072 ' 反馈报警未激活 '

检测时间： 机床参数确认。
引起原因： 轴的反馈报警未激活。
解决方案： 设置轴参数 FBACKAL = TRUE。

0073 ' 软件极限未激活 '

检测时间： 机床参数确认。
引起原因： 软件极限未激活。轴参数 LIMIT+ 和 LIMIT- 设置为 0。
解决方案： 将轴参数 LIMIT+ 和 LIMIT- 设置成轴极限值。

0074 ' 趋势测试未激活 '

检测时间： 机床参数确认。
引起原因： 趋势测试未激活。该测试核对轴是否在期望方向运动。
解决方案： 将轴参数 TENDENCY 设为 TRUE。

0075 'I/O 配置表错误 '

检测时间： 机床参数确认。
引起原因： 参数 NDINPUT 或 NDOUTPUT 必须等于硬件检测到的 I/O 模块数。
解决方案： 改变机床参数 NDINPUT 或 INDOUTPUT 的值。

0076 ' 每个通道的轴或主轴总数超过轴或主轴的总数 '

检测时间： 机床参数确认。
引起原因： 参数CHNAXIS或CHNSPDL (每个通道的轴和主轴数量)大于参数NAXIS或NSPDL (系统的轴和主轴数量) 的值。
解决方案： 改变 CHNAXIS 或 CHNSPDL 的值。

0077 ' 轴或主轴分配给了一个以上的通道 '

检测时间： 机床参数确认。
引起原因： 轴或主轴分配给了多个通道。
解决方案： 核对所有通道的 CHAXISNAME 和 CHSPDLNAME 参数表。

0078 ' 主轴和从轴必须从属于同一个通道 '

检测时间： 机床参数确认。
引起原因： 龙门对由不同通道的轴组成。
解决方案： 核对龙门轴表以及用于为通道分配轴的 CHAXISNAME 表。

0079 ' 不能停止龙门从轴 '

检测时间： 机床参数或启动过程确认。
引起原因： 龙门从轴的 “PARKED” 激活。在 CNC 上电时，检测到龙门对中的轴停止。
解决方案： 别停止轴或取消龙门对。

0080 ' 要让轴生效，先让通用参数表生效 '

检测时间： 机床参数确认。
引起原因： 通用参数表的 NAXIS 值增加，且在该表未生效的情况下，试图让系统中某根新轴的表生效。
解决方案： 轴参数生效前，先让轴的通用参数表生效。由于它可检测 NAXIS 中的变化，所以要解决该问题，重启 CNC。



0081 ' 适当位置区域不能小于轴的分辨率 '

检测时间： 机床参数确认。

引起原因：

解决方案： 增加轴的适当位置区域，即轴设置表格中的 INPOSW 参数。

0082 ' 螺线误差或交叉补偿不完全适合轴 '

检测时间： 机床参数确认。

引起原因： 旋转模块轴在设置和螺线补偿中具有不同的机床参数 MODUPLIM/MODLOWLIM，或者作为施加影响的轴 (MOVAXIS) 而属于交叉补偿的一部分。

解决方案： 为龙门对的所有 SET（ 设置 ） 设置相同的模块极限。如果该方法不能解决问题，则没有其他解决方案。

0083 ' 主轴和从轴类型必须相同 (DRIVEAXIS)'

检测时间： 机床参数确认。

引起原因： 龙门轴的轴驱动类型不同 (SERCOS/ 模拟)。

解决方案： 核对机床参数。为两根轴分配相同类型的驱动 (DRIVEAXIS)。

0084 ' 如果通道的轴或主轴不能交换，要为其分配名称 '

检测时间： 机床参数确认。

引起原因： 必须为通道的主轴或轴指定名称，即令参数 AXISEXCH = NO(可参见通道的 CHAXISNAME 或 CHSPDLNAME 表)。

解决方案： 为通道分配轴或主轴 (CHAXISNAME 或 CHSPDLNAME)。

0085 ' 系统中没有 SERCOS 轴 '

检测时间： 机床参数确认。

引起原因： 相关参数表中定义了变量 SERCOS，但是系统中没有 SERCOS 轴。

解决方案： 去掉定义的 SERCOS 变量。

0086 ' 不是 SERCOS 轴 '

检测时间： 机床参数确认。

引起原因： 为非 SERCOS 轴定义了 SERCOS 变量。

解决方案： 去掉为该轴定义的该变量。

0087 'SERCOS 变量过多 '

检测时间： 机床参数确认。

引起原因： 超出 SERCOS 变量的最大数目 (100)。

解决方案： 减少变量数。

0088 ' 内部变量跟踪被激活 '

检测时间： 机床参数确认。

引起原因： 正执行内部变量跟踪。

解决方案： 联系 Fagor 公司。

0089 ' 机床参数出错，导致启动时只有一个通道 '

检测时间： 机床参数确认。

引起原因： 在确认与通道轴或主轴相关的参数时，出现错误或警告。

例如，通道拥有一根与其相关的轴 (CHAXISNAME)，但是轴并不在系统轴表 (AXISNAME) 中。由于不能配置 n 个通道， CNC 只能在配置一个通道时启动。

解决方案： 改变机床参数，防止出现与其他机床参数相关的警告或错误。该警告可以消除，而不必更改 NCHANNEL 参数 (通道号)。

0090 ' 机床参数错误导致系统带缺省轴配置启动 '

检测时间： 机床参数确认。

引起原因： 在确认与通道轴或主轴相关的参数时，出现错误或警告。

例如，轴的数量 (NAXIS) 大于包含轴名 (AXISNAME) 的列表中的相关项。由于不能在用户配置下进行启动， CNC 只能在缺省配置下启动。

解决方案： 改变机床参数，防止出现与其他机床参数相关的警告或错误。



0091 ' 标识符 (ID) 相同的 SERCOS 变量不能拥有不同的记忆存储器 '

检测时间： 机床参数确认。
引起原因： 在机床参数 (OEM、DRIVEVAR、DATA) 的变量表中，某些变量名称 (MNEMONIC) 相同，但 SERCOS 标识符 (ID) 不一样。
解决方案： 为相同标识符 (ID) 的 SERCOS 变量指定相同的名称。

0092 ' 标识符 (ID) 相同的 SERCOS 变量不能拥有不同的 MODE 或 TYPE'

检测时间： 机床参数确认。
引起原因： 在机床参数 (OEM、 DRIVEVAR、 DATA) 的 SERCOS 变量表中，某些变量标识符 (ID) 相同，但是 TYPE (同步或异步) 或 MODE (读或写) 不同。
解决方案： 为相同标识符 (ID) 的 SERCOS 变量指定相同的 MODE 和 TYPE 。

0093 ' 名称 (MNEMONIC) 相同的 SERCOS 变量不能拥有不同的 ID、 MODE 或 TYPE'

检测时间： 机床参数确认。
引起原因： 在机床参数 (OEM、 DRIVEVAR、 DATA) 的 SERCOS 变量表中，某些变量名称相同，但是 TYPE (同步或异步) 、 MODE (读或写) 或 SERCOS 标识符 (ID) 不同。
解决方案： 为名称 (MNEMONIC) 相同的 SERCOS 变量指定相同的 MODE、 TYPE 和 ID 。

0100 ' 等待报告的变量太多 '

检测时间： 执行程序时。
引起原因： 已被修改且必须报至界面的变量数超出最大允许数量。
解决方案： 按 [ESC] 键。

0104 ' 通讯中止 '

检测时间： 执行程序时。
引起原因： 对外部变量的读 / 写操作尚未结束。
解决方案： 请与厂商联系。

0105 ' 执行程序过程中不能验证参数 '

检测时间： 执行程序时。
引起原因： 在程序运行或中断时，试图验证机床参数表。
解决方案： 结束或中止程序，以验证机床参数表。

0150 ' 打开文件太多 '

检测时间： 使用外部子程序运行工件程序时。
引起原因： 打开的文件 (主程序与子程序) 数大于 20。
解决方案： 减少同一时间在工件程序中打开的外部子程序数量。

0151 ' 写操作被拒绝 '

检测时间： 存取文件时。
引起原因： 试图对没有写操作权限的文件进行写操作。
解决方案： 赋予文件写操作权限。

0152 ' 不能打开文件 '

检测时间： 存取文件时。
引起原因： 不能打开文件进行读写操作。
解决方案： 检查文件是否位于该目录下，并且具有相应的操作权限。
检查文件是否被前面的某些操作所破坏。

0153 ' 读操作被拒绝 '

检测时间： 存取文件时。
引起原因： 试图对没有读操作权限的文件进行读操作。
解决方案： 赋予文件读操作权限。



0160 ' 系统的轴 / 设置不可用 '

检测时间：　执行 SET AX、 CALL AX、 G112 指令时。

引起原因：　在变更轴配置或参数设置时，试图对不存在的轴或参数设置、或者当前与另一通道关联的轴进行访问。

解决方案：　不访问不可用的轴或参数设置，或者使用 FREE AX 指令释放另一通道中的轴。

0168 'LR 溢出 '

检测时间：　SERCOS 轴的位置环中。

引起原因：　SERCOS 轴的位置环超出允许的时间。

解决方案：　调整 LOOPTIME 参数。

0200 ' 请求 VxD 失败 '

检测时间：　读取电池状态时。

引起原因：　不能与 VcompciD 联络。

解决方案：　请与厂商联系。

0201 ' 停电，PC 机由电池供电 '

检测时间：　读取电池状态时。

引起原因：　PC 机主电源掉电，正由电池供电。

解决方案：　如果停电偶尔发生，请查找可能的原因。如果是操作者导致掉电，则可让自动 PC 支路语句结束运行。

不管停电是偶尔的还是由操作者引起的，均结束运行自动 PC 支路语句。如果停电偶尔发生，请查找可能的原因。

1000-1999 号错误

1000 ' 指令功能需要对轴进行编程 '

检测时间： 执行程序时。
引起原因： 未对受编程指令或 G 功能影响的轴进行编程。
解决方案： 核对程序。

1004 ' 主轴转速为 0'

检测时间： 执行程序时。
引起原因： G63 功能激活时，为其编写的主轴转速为 0。
解决方案： 编写主轴转速 S。

1005 ' 运动模块的进给率为 0'

检测时间： 执行程序时。
引起原因： 未为运动轴编写进给率 (F)。
解决方案： 编写进给率 F。

1006 'G20: 不允许对主轴编程 '

检测时间： 执行程序时。
引起原因： G20 功能不允许对主轴进行编程。
解决方案： 核对程序。

1007 ' 编程功能需要一根不存在的主轴 '

检测时间： 执行程序时。
引起原因： 下述功能需要主平面内的两根轴：
- G2、 G3、 G8、 G9、 G30、 G36、 G37、 G38、 G39 以及 G73。
- G20：当冲突检测激活时。
- 激活冲突检测：G20。

下述功能需要主平面内的一根轴：G11、 G12、 G13 以及 G14。
解决方案： 核对程序。

1008 ' 坐标超出规定范围 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
- 使用 G101 功能时，轴中包含的测量偏移量太大。
- 轴的编程坐标太大。

解决方案： 核对程序。

1009 'G4: 编写两次暂停：直接暂停和使用 K 的暂停 '

检测时间： 执行程序时。
引起原因： 在同一程序段中编写了两次 G4 暂停，直接使用数字的暂停以及使用 K 的暂停。
解决方案： 只编写一次 G4 暂停功能。



1010 ' 编写 G4 K'

检测时间： 执行程序时。

引起原因： 功能编程错误。

解决方案： 使用 G4 编写暂停时可使用两种方法：

- G4 < 时间 >
- G4 K< 时间 >

两种情况下，暂停时间都必须位于 G4 后。

第二种情况下不能在 "K" 后加 "="。

1011 'G4: 暂停时间超出规定范围 '

检测时间： 执行程序时。

引起原因： 暂停功能 G4 的编程值过大。

解决方案： 最大允许暂停值为 2147483646。

1012 'G4: 不能使用 K 编写暂停 '

检测时间： 执行程序时。

引起原因： 字母 K 与通道的第三根轴关联，这种情况下，将没有第三根轴。

解决方案： 如果通道不希望有第三根轴，可直接通过数字编写暂停。

1013 'G4: 暂停时间不能为负 '

检测时间： 执行程序时。

引起原因： 使用 G4 功能或编写 #TIME(时间) 指令时，暂停时间为负。

解决方案： 暂停时间必须等于或大于 0。

1014 ' 曲面轴在镜像时不能以直径进行编程 '

检测时间： 执行程序时。

引起原因： 曲面轴 (机床参数 FACEAXIS = Yes) 不能同时激活镜像与直径编程。

解决方案： 激活镜像或者直径编程。

1015 ' 圆心坐标超出规定范围 '

检测时间： 执行程序时。

引起原因： 圆弧插补圆心或旋转中心的 I、 J、 K 值过大。

解决方案： 编写较小的值。

1016 ' 以直径形式为轴编程时不能使用负值 '

检测时间： 执行程序时。

引起原因： 以直径形式 (机床参数 DIAMPROG) 进行绝对坐标编程时 (G90 功能)，为轴编写了负坐标。

解决方案： 以绝对坐标编程时，轴的直径必须为正。

1017 'G198: 负软件极限超出规定范围 '

检测时间： 执行程序时。

引起原因： 编写的负软件极限值过高。

解决方案： 核对程序。

1018 'G199: 正软件极限超出规定范围 '

检测时间： 执行程序时。

引起原因： 编写的正软件极限超出规定范围。

解决方案： 核对程序。

1019 ' 未对被请求的轴进行测量 '

检测时间： 执行程序时。

引起原因： 使用 G101 功能对轴的偏移量进行测量，该轴：

- 此前未进行测量 (G100)。
- 正在进行的测量 (G100) 被取消 (G102)。

解决方案： 在使用 G101 功能的轴上进行测量。



1020 ' 斜升时间为负 '

检测时间： 执行程序时。
引起原因： 使用 G132 功能时，编写了负的斜升时间。
解决方案： 斜升时间必须等于或大于 0。

1021 ' 斜升时间超出规定范围 '

检测时间： 执行程序时。
引起原因： G132 功能编写的斜升时间过长。
解决方案： 核对程序。

1022 ' 前馈百分比超出规定范围 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- G134 功能编写的前馈百分比过大。
- G135 功能编写的 交流前馈百分比过大。

解决方案： 前馈或交流前馈的百分比必须大于 0 且小于 120。

1023 ' 设置号错误 '

检测时间： 执行程序时。
引起原因： 轴的编程设置号错误。
解决方案： 轴的编程设置必须大于 0 且小于或等于轴的机床参数 NPARSETS。

1024 ' 设置号超出规定范围 '

检测时间： 执行程序时。
引起原因： 编写的参数设置值过大。
解决方案： 设置号的最大许可值为 65535。

1025 ' 编程距离为 0'

检测时间： 执行程序时。
引起原因： G63 程序段中未编写运动。
解决方案： 编写运动。

1026 ' 编程半径导致错误圆周轨迹 '

检测时间： 执行程序时。
引起原因： 圆弧插补的编程半径太小。
解决方案： 核对程序。

1027 ' 圆周轨迹的起点与终点相同 (无穷多的可能性)'

检测时间： 执行程序时。
引起原因： 圆弧插补的半径为 0，从而有无穷多个可能性。
解决方案： 核对程序。

1028 ' 编程圆心与计算出的圆心差值过大 '

检测时间： 执行程序时。
引起原因： 在圆弧插补中，当 265 功能有效时，初始半径与最终半径的差超出机床参数 CIRINERR 与 CIRINFACT 的值。
解决方案： 正确编写圆弧插补。

1029 ' 圆周轨迹半径为 0'

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 圆弧插补的编程半径为 0。
- G265 功能有效时， CNC 根据圆弧插补的圆心坐标计算出的半径为 0。
- G265 功能无效时，圆弧插补的编程半径为空。

解决方案： 圆弧插补的半径不能为 0。两个圆心坐标均不能为空。



1030 '#AXIS(轴) 未编写 G200/G201/202'

检测时间： 执行程序时。
引起原因： 指令 #AXIS（轴）的同一行未编写 G200/201/202。
解决方案： 在同一行同时编写两种功能。

1031 ' 需要 #AXIS（轴）'

检测时间： 执行程序时。
引起原因： G201 功能的同一行里未编写 #AXIS（轴）指令。
解决方案： 在同一行编写两种功能。

1032 'M19 没有主轴位置信息 '

检测时间： 执行程序时。
引起原因： 编写了 M19 功能，但没有编写主轴位置。
解决方案： 核对程序。

1035 '#SLOPE: 参数超出规定范围 '

检测时间： 执行程序时。
引起原因： 参数的编程值过大。
解决方案： 编写较小的值。

1037 'G0/G1/G100/G63 激活时将忽略圆心坐标 '

检测时间： 执行程序时。
引起原因： I、 J、 K 的编程值将被忽略。
解决方案： 以上功能并不需要对这些参数进行编程。

1038 ' 测量时不能激活半径补偿 '

检测时间： 执行程序时。
引起原因： 半径补偿激活时试图执行 G100， G41-G42。
解决方案： 核对程序。

1039 ' 轴已经有测量值 '

检测时间： 执行程序时。
引起原因： 试图使用 G100，但前面的 G101 已经包含测量偏置。
解决方案： 轴包含的偏置可由 G102 删除。

1040 'G201 中的轴不允许进行原点搜索 '

检测时间： 执行程序时。
引起原因： 轴不能在附加手动模式 (G201) 中回零。
解决方案： 要对轴进行回零，可使用 G202 功能取消它的附加手动模式。若有必要，可重新激活 G201 功能。

1041 ' 修正后的圆周轨迹中心超出规定范围 '

检测时间： 执行程序时。
引起原因： 使用以下方式定义了圆弧插补：
- 使用半径 "R" 或 "R1" 以及终点坐标。
- G265 功能 (圆心修正) 激活时，使用中点与终点的坐标。

CNC 计算出来的插补圆心坐标值过大。
解决方案： 程序段中为插补半径编写的中点或终点的值太大。

1043 ' 平面的第三根轴不能与第一或第二根轴相同 '

检测时间： 执行程序时。
引起原因： G20 功能 (改变平面) 的参数 5 与参数 1 或 2 相同。
解决方案： 如果刀具的纵轴 (参数 3) 与平面的第一或第二根轴相同 (参数 1 和 2)，必须使用参数 5 对第三根轴进行编程。该参数不能与第一或第二个参数相同。



1044 ' 平面的第一和第二根轴不能相同 '

检测时间: 执行程序时。
引起原因: 使用 G20 功能 (改变平面) 时，平面的第一根轴 (参数 1) 和第二根轴 (参数 2) 相同。
解决方案: 核对程序。

1045 ' 平面的第一根轴编程错误 '

检测时间: 执行程序时。
引起原因: 使用 G20 功能 (改变平面) 时，平面的第一根轴 (参数 1) 编程错误。
解决方案: 平面的第一根轴必须为配置的三根主轴之一。

1046 ' 平面的第二根轴编程错误 '

检测时间: 执行程序时。
引起原因: 使用 G20 功能 (改变平面) 时，平面的第二根轴 (参数 2) 编程错误。
解决方案: 平面的第二根轴必须为配置的三根主轴之一。

1047 ' 平面需要第三根轴 (索引 5) '

检测时间: 执行程序时。
引起原因: 与 G20 功能 (改变平面) 编写的参数 5 丢失或错误。
解决方案: 如果刀具的纵轴 (参数 3) 与平面的第一或第二根轴相同 (参数 1 和 2)，必须使用参数 5 编写第三根轴，它不能与第一或第二个参数相同，且必须为配置的三根主轴之一。

1048 ' 刀具长度补偿的半径超出规定范围 '

检测时间: 执行程序时。
引起原因: 刀具尺寸超出最大值。
解决方案: 修正刀具尺寸。

1049 ' 在活动平面内重复定义曲面轴 (FACEAXIS)'

检测时间: 执行程序时。
引起原因: 主平面内的两根轴均为曲面轴。且它们的机床参数 FACEAXIS = Yes。
解决方案: 工作平面内只能有一根曲面轴。

1050 ' 查看刀具偏置，该值超出规定范围 '

检测时间: 执行程序时。
引起原因: 刀具尺寸超出最大值。
解决方案: 修正刀具尺寸。

1052 ' 测量得出的值超出规定范围 '

检测时间: 执行程序时。
引起原因: 使用 G100 功能进行测量时，得到的值大于探测点坐标或获得的探测运动偏置。
解决方案: 探测运动中获得的值应该介于 (-2147483647、 2147483646) 之间。

1054 ' 夹具不存在 '

检测时间: 执行程序时。
引起原因: 编写的夹具索引错误。
解决方案: 夹具索引值必须大于 0 且小于 10。

1055 ' 不能在同一程序段中同时修正 D 和刀具半径 '

检测时间: 执行程序时。
引起原因: 试图在编写了自动换刀和刀具偏置变化的程序段中写入 "V.G.TOR" 变量。
解决方案: 编写在不同的程序行中。

1056 ' 外部变量过多 '

检测时间: 执行程序时。
引起原因: 外部变量达到 CNC 许可的最大值 (500) 。
解决方案: 核对程序。

1057 ' 变量没有读取权限 '

检测时间： 执行程序时。

引起原因： 试图从零件程序或 MDI 中读取没有读操作权限的变量。

解决方案： 该变量不能从零件程序或 MDI 中读取。可通过编程手册查询每个变量是否具有从程序 / MDI、界面和 PLC 中进行读取操作的权限。

1059 ' 变量没有写操作权限 '

检测时间： 执行程序时。

引起原因： 用户试图对没有写操作权限的变量进行写操作。试图从零件程序或 MDI 中写没有写操作权限的变量。

解决方案： 该变量不能写。不能通过零件程序或 MDI 对该变量进行写操作。

可通过编程手册查询每个变量是否具有从 程序 / MDI、界面和 PLC 中进行写操作的权限。

1060 'N 标号值超出规定范围 '

检测时间： 执行程序时。

引起原因： 定义了一个不在许可范围内的 "N" 程序段号。

解决方案： 程序段号必须为小于 2147483646 的正值。

1061 'G 功能不存在 '

检测时间： 执行程序时。

引起原因： 编写的 G 功能不存在。

解决方案： 核对程序。

1062 'G 功能不相容 '

检测时间： 执行程序时。

引起原因： 在同一程序段中编写了两个不相容的 G 功能。

解决方案： 将它们编写在不同的程序段中。

1063 ' 不相容的 G 功能 (G108/G109/G193)'

检测时间： 执行程序时。

引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。

解决方案： 将它们编写在不同的程序段中。

1064 ' 不相容的 G 功能 (G196/G197)'

检测时间： 执行程序时。

引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。

解决方案： 将它们编写在不同的程序段中。

1065 ' 不相容的 G 功能 (G17/G18/G19/G20)'

检测时间： 执行程序时。

引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。

解决方案： 将它们编写在不同的程序段中。

1066 ' 不相容的 G 功能 (G136/G137)'

检测时间： 执行程序时。

引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。

解决方案： 将它们编写在不同的程序段中。

1067 ' 不相容的 G 功能 (G40/G41/G42)'

检测时间： 执行程序时。

引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。

解决方案： 将它们编写在不同的程序段中。

1068 ' 不相容的 G 功能 (G151/G152)'

检测时间： 执行程序时。

引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。

解决方案： 将它们编写在不同的程序段中。

1069 ' 不相容的 G 功能 (G54-G59/G159)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1070 ' 不相容的 G 功能 (G5/G7/G50/G60/G61)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1071 ' 不相容的 G 功能 (G70/G71)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1072 ' 不相容的 G 功能 (G80-G88/G160-G166/G281-G286/G287-G297)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1073 ' 不相容的 G 功能 (G90/G91)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1074 ' 不相容的 G 功能 (G93/G94/G95)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1075 ' 不相容的 G 功能 (G96/G97/G192)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1077 ' 不相容的 G 功能 (G115/G116/G117)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

11079 ' 不相容的 G 功能 (G138/G139)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1080 ' 不相容的 G 功能 (G6/G261/G262)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1081 ' 不相容的 G 功能 (G264/G265)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1082 ' 不相容的 G 功能 (G200/G201/G202)'

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1084 ' 刀具半径补偿激活时不允许改变平面 '

检测时间： 执行程序时。
引起原因： 在刀具半径补偿激活时，使用 G17-G20 (改变平面) 功能改变平面的第一根或第二根轴。
解决方案： 取消补偿，定义新的工作平面。

1085 ' 有效平面的第一或第二根轴缺失时，不允许使用 G41/G42'

检测时间： 执行程序时。
引起原因： 如果通道缺失有效平面两根轴中的一根，则不能进行刀具半径补偿。
解决方案： 使用 #CALL AX 或 #SET AX 指令恢复缺失的轴。

1087 ' 需要 "="'

检测时间： 执行程序时。
引起原因： 编写的指令或功能语法错误。
解决方案： 查询编程手册。核对程序。

1088 ' 偏置号错误 '

检测时间： 执行程序时。
引起原因： 使用高级语言编程时，为 G 功能的参数赋予了错误的值。
解决方案： G159 功能的许可值范围为 1 ～ 20。

1089 ' 不相容的 M 功能 (M3/M4/M5/M19)'

检测时间： 执行程序时。
引起原因： 在同一程序段中，为同一主轴编写了一个以上的 M 功能 (M3/M4/M5/M19) 。
解决方案： 在不同的程序段中为同一主轴编写 M 功能。

1090 'H 功能不存在 '

检测时间： 执行程序时。
引起原因： 编写的 H 功能不存在。
解决方案： 最大的 H 功能号为 65534。

1091 ' 重复编写 T 功能 '

检测时间： 执行程序时。
引起原因： 同一程序段中编写了一个以上的 T 功能。
解决方案： 将它们编写在不同的程序段中。

1093 ' 重复编写 D 功能 '

检测时间： 执行程序时。
引起原因： 同一程序段中编写了一个以上的 D 功能。
解决方案： 将它们编写在不同的程序段中。

1094 ' 重复编写 F 进给率 '

检测时间： 程序执行时。
引起原因： 同一程序段中编写了一个以上的 F 功能。
解决方案： 将它们编写在不同的程序段中。

1095 ' 进给率 F 不能为负值或 0'

检测时间： 执行程序时。
引起原因： 进给率 (F) 必须为正且不能为 0。
解决方案： 核对程序。

1096 ' 不能使用 E 编写进给率 '

检测时间： 执行程序时。
引起原因： 使用字母 E 编写轴进给率。
解决方案： 使用 F 编写轴进给率。



1097 ' 未知的主轴名 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
- 主轴名错误。
- 主轴名有效，但是不存在于系统中。
- 主轴名存在于系统，但是它不属于该通道。

解决方案： 有效的主轴名为 S、 S1、 . . . 、 S9。
程序段中涉及的主轴必须存在于系统配置中，且依赖于通道配置中的指令。

1098 ' 重复编写 S 主轴转速 '

检测时间： 执行程序时。
引起原因： 在同一程序段中为同一主轴编写了一个以上的 S 功能。
解决方案： 将它们编写在不同的程序段中。

1100 ' 参数索引超出规定范围 '

检测时间： 执行程序时。
引起原因： 所访问的算术参数的索引超出相关机床参数确定的范围。
- 局部参数的 MINLOCP-MAXLOCP。
- 全局参数的 MINGLBP-MAXGLBP。
- 通用参数的 MINCOMP-MAXCOMP。

解决方案： 解决办法有：
- 访问索引在规定范围内的参数。
- 修正当前机床参数并重启 CNC，使新值生效。

1101 '#SET IPOPOS 指令编程错误 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于前一种情况，查阅编程手册。对于第二种情况，可在不同的程序段中编写。

1102 'R 的索引只能为 1'

检测时间： 执行程序时。
引起原因： 半径只能使用 R 或 R1。
解决方案： 核对程序。

1103 'O 功能不存在 '

检测时间： 执行程序时。
引起原因： 编写的 O 功能不存在。
解决方案： 核对程序。

1104 ' 主程序中不允许使用 "%" 号 '

检测时间： 执行程序时。
引起原因： "%" 号只能作为定义主程序或局部子程序名的第一个字符使用。
解决方案： 从程序中删除该符号。

1105 ' 需要赋值算子 '

检测时间： 执行程序时。
引起原因： 变量或参数后未编写赋值算子。
解决方案： 有效的赋值算子包括 "="、 "+="、 "-="、 "*="、 "/="。

1106 ' 需要 "]"'

检测时间： 执行程序时。
引起原因： 表达式或指令中缺少 "]"（方括弧）。
解决方案： 检查程序段的语法。



1107 ' 轴不存在或无效 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- 试图移动一根不存在或在系统或通道中无效的轴。
- 在高级语句中编写的轴不存在或在系统或通道中无效。
- 在轴变量中，编写了不存在于系统中的轴号。
- 在轴变量中，编写了不存于通道中的轴索引。

解决方案： 确定编写的轴存在于系统或通道中且可用 (未停车)。

1108 ' 轴重复编程 '

检测时间： 执行程序时。

引起原因： 轴跟以下功能中的一种，在同一程序段中进行了多次编程：

- 轴运动， G0、 G1、 G2、 G3、 G8 或 G9。
- 加工螺纹， G33 或 G63。
- 指令 #FACE 或 #CYL。
- 释放平面选择， G20。

与轴运动相关的功能，也有可能是同时编写了笛卡儿坐标和极坐标导致的。

解决方案： 核对程序。

1109 ' 轴索引错误 '

检测时间： 执行程序时。

引起原因： G20 和 G74 功能中，轴名必须编写索引。该索引错误。

解决方案： 轴索引必须大于 0，且不能超过系统或通道中轴的最大数量。

1110 ' 重复编写 I、 J、 K 值 '

检测时间： 执行程序时。

引起原因： 程序段中的 I、 J、 K 值重复编写。

解决方案： 在程序段中只编写一次 I、 J、 K 值。

1111 ' 控制指令 $ 必须位于单独的程序段中 '

检测时间： 执行程序时。

引起原因： 只有标签的程序段号可以与 "$" 指令编写在同一程序段中。
唯一的例外是在同一程序段中编写 $IF 和 $GOTO 指令。

解决方案： 在不同程序段中编程。

1112 '$IF 指令 < 条件 > 后只能跟 $GOTO'

检测时间： 执行程序时。

引起原因： 控制指令必须编写在单独行中。唯一的例外就是在同一行中编写 $IF 和 $GOTO。

解决方案： 将它们编写在不同的程序段中。

1113 '$ELSE 指令多余 '

检测时间： 执行程序时。

引起原因： 程序段中编写了 $ELSE 指令，但是前面并没有 $IF 指令。

解决方案： 核对程序。

1114 '$ELSE 指令必须位于单独的程序段中 '

检测时间： 执行程序时。

引起原因： 控制指令必须位于单独程序段中。

解决方案： 核对程序。

1115 '$ELSEIF 指令多余 '

检测时间： 执行程序时。

引起原因： 程序段中编写了 $ELSEIF 指令，但是前面并没有 $IF 指令。

解决方案： 核对程序。

1116 '$ELSEIF < 条件 > 指令必须位于单独的程序段中 '

检测时间： 执行程序时。

引起原因： 控制指令必须位于单独程序段中。

解决方案： 核对程序。



1117 '$ENDIF 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $ENDIF 指令，但是前面并没有 $IF 指令。
解决方案： 核对程序。

1118 '$ENDIF 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1119 '$SWITCH < 表达式 > 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1120 '$CASE 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $CASE 指令，但是前面并没有 $SWITCH 指令。
解决方案： 核对程序。

1121 '$CASE < 表达式 > 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1122 '$DEFAULT 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $DEFAULT 指令，但是前面并没有 $SWITCH 指令。
解决方案： 核对程序。

1123 '$DEFAULT 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1124 '$ENDSWITCH 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $ENDSWITCH 指令，但是前面并没有 $SWITCH 指令。
解决方案： 核对程序。

1125 '$ENDSWITCH 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1126 '$FOR: 计数变量无效 '

检测时间： 执行程序时。
引起原因： $FOR 控制循环中使用了无效计数器。
解决方案： 只有变量与算术参数才能用作 $FOR 控制循环的有效计数器。

1127 '$FOR 指令 < 条件 > 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1128 '$FOR: 条件中的字符过多 '

检测时间： 执行程序时。
引起原因： $FOR 指令所在行的字符总数超过 5100。
解决方案： 缩减定义 $FOR 指令的行。



1129 '$ENDFOR 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $ENDFOR 指令，但是前面并没有 $FOR 指令。
解决方案： 核对程序。

1130 '$ENDFOR 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1131 '$WHILE < 条件 > 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1132 '$WHILE: 条件中的字符过多 '

检测时间： 执行程序时。
引起原因： 控制循环 $WHILE 条件中的字符超出最大许可值。
解决方案： 允许使用的字符最多为 5000 个。

1133 '$ENDWHILE 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $ENDWHILE 指令，但是前面并没有 $WHILE 指令。
解决方案： 核对程序。

1134 '$ENDWHILE 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1135 '$DO 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1136 '$ENDDO 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $ENDDO 指令，但是前面并没有 $DO 指令。
解决方案： 核对程序。

1137 '$ENDDO < 表达式 > 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1138 '$BREAK 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 控制指令必须位于单独程序段中。
解决方案： 核对程序。

1139 '$BREAK 指令多余 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $BREAK 指令，但是没有打开控制循环：$IF、 $ELSE、 $FOR、 $WHILE、 $DO 或 $CASE。
解决方案： 核对编写的所有 "$" 控制循环。$BREAK 指令用来结束 $CASE 或者从 $IF、$ELSE、 $WHILE、 $FOR 或 $DO 循环中跳出。



1140 '$CONTINUE 指令多余 '

检测时间： 执行程序时。

引起原因： 没有打开控制循环， $CONTINUE 指令无意义。

解决方案： 核对程序段的顺序。

$CONTINUE 指令编写在控制循环 $FOR、$WHILE 或 $DO 的内部，以返回循环起点。

1141 '$CONTINUE 指令必须位于单独的程序段中 '

检测时间： 执行程序时。

引起原因： 控制指令必须位于单独程序段中。

解决方案： 核对程序。

1142 '#TIME 指令必须位于单独的程序段中 '

检测时间： 执行程序时。

引起原因： 暂停必须与 #TIME 指令编写在同一程序段中。

解决方案： 核对程序。

1146 'G37 前的轨迹必须为直线 '

检测时间： 执行程序时。

引起原因： 切入前的运动程序段轨迹必须为直线。

解决方案： 核对程序。

1147 'G38 后的轨迹必须为直线 '

检测时间： 执行程序时。

引起原因： 切出后的运动程序段轨迹必须为直线。

解决方案： 核对程序。

1149 'G36/G37/G38/G39 未执行 '

检测时间： 执行程序时。

引起原因： 不能以编程半径实现第一和最后程序段之间的连接轨迹。

解决方案： 核对连接轨迹的编程半径值。确保结合处有可能连接第一和最后程序段。

1150 'G36/G37/G38/G39 功能后必须跟运动程序段 '

检测时间： 执行程序时。

引起原因： 连接轨迹的第二个运动程序段缺失。

解决方案： 在定义连接轨迹的 G 功能与第二运动程序段之间不要编写任何程序段。

1151 'G8/G36/G37/G38/G39 功能前必须有运动程序段 '

检测时间： 执行程序时。

引起原因： 连接轨迹的第一运动程序段缺失。

解决方案： 在定义连接轨迹的 G 功能与第一运动程序段之间不要编写任何程序段。

1152 ' 嵌套子程序过多 '

检测时间： 执行程序时。

引起原因： 超过了 CNC 允许的最大嵌套层。

可以从主程序 (或子程序) 调用子程序，该子程序也可调用其它子程序，以此类推。CNC 规定最大调用次数为 20 层嵌套。

解决方案： 重新设计程序，减少子程序 (局部和全局) 调用嵌套层。

1153 ' 程序中定义的局部子程序过多 '

检测时间： 执行程序时。

引起原因： 局部子程序超出了 CNC 许可的最大值。最大值为 100。

解决方案： 重新设计程序，减少定义的局部子程序数目。



1154 ' 文件名过长 '

检测时间： 执行程序时。

引起原因： 程序或子程序名的字符数超过了最大许可值。

- 程序或子程序名最长为 63 个字符。
- 程序或子程序的路径最长为 120 个字符。

编写带路径的程序或子程序名时，最大字符数为两者之和。

解决方案： 减少主程序或子程序名的字符数。将程序或子程序移动到另一个目录下，以减少路径的字符数。

1155 ' 不能访问文件 '

检测时间： 执行程序时。

引起原因： 不能访问程序或子程序。

解决方案： 确认文件有效且未被破坏。

调用子程序时，检查文件名与路径是否正确。

如果搜索语句中没有明确指示子程序路径，那么它将以以下顺序执行：

1. #PATH 指令编写的路径。
2. 正在执行的程序的路径。
3. 机床参数 SUBPATH 的路径。

1156 ' 未找到主程序 '

检测时间： 执行程序时。

引起原因： 未找到主程序。

解决方案： 文件中若定义了局部子程序，主程序必须以 "% 程序名 " 开始。

1157 ' 未找到全局子程序 '

检测时间： 执行程序时。

引起原因： 程序调用的全局子程序未找到。

解决方案： 检查文件名与路径是否正确。

如果搜索语句中没有明确指示子程序路径，那么它将以以下顺序执行：

- #PATH 指令编写的路径。
- 正在执行的程序的路径。
- 机床参数 SUBPATH 的路径。

1159 ' 局部子程序名过长 '

检测时间： 执行程序时。

引起原因： 局部子程序名的最大字符数为 63。

解决方案： 核对程序。

1160 ' 未找到局部子程序 '

检测时间： 执行程序时。

引起原因： 未找到欲执行的局部子程序。

解决方案： 确定调用程序段中的局部子程序名与定义的名字一样。

局部子程序在文件的前端进行定义。其定义以%L开始，以#RET、M17或M29结束。

1161 '$ 控制程序段开启 '

检测时间： 执行程序时。

引起原因： 在子程序中， "$" 控制程序段没有对应的关闭指令。

解决方案： 检查控制程序段，并正确关闭。

1162 'M17/M29/#RET 指令多余 '

检测时间： 执行程序时。

引起原因： 检测到 M17/M29/#RET 作为结束主程序的指令使用。

解决方案： 编写 M30/M02，以结束主程序。

如果错误仍然存在，则确定所有的局部子程序 (%L) 与全局子程序均以 M17/M29/#RET 结束。



1163 'M30/M02 指令多余 '

检测时间： 执行程序时。
引起原因： 检测到子程序的末尾使用了 M30 或 M02。
解决方案： 确定所有的局部子程序 (%L) 与全局子程序均以 M17/M29/#RET 结束。

1164 ' 数学表达式存在未知术语 '

检测时间： 执行程序时。
引起原因： 数学表达式中编写的术语存在控制语法错误。
解决方案： 检查表达式的所有术语：变量、参数、算子等。

1165 ' 变量不存在 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 请求的变量不存在。
- 变量名语法错误。
- 变量为数组，而数组索引未指明。
- 特殊轴或以其他方式请求通用变量。

解决方案： 核对程序。

1166 ' 负数开平方根 '

检测时间： 执行程序时。
引起原因： 在数学表达式中，求负数的平方根 "SQRT"。
解决方案： 核对程序。

1167 ' 负数或 0 求对数 '

检测时间： 执行程序时。
引起原因： 在数学表达式中，求负数或 0 的对数 "LOG" 或 "LN"。
解决方案： 核对程序。

1168 ' 变量索引超出规定范围 '

检测时间： 执行程序时。
引起原因： 数组变量请求的索引错误。
解决方案： 数组变量容许的最小索引为 1，最大索引则取决于变量本身。有些可容许索引为 0 的特例：G.GS、 G.MS、 G.LUP1、 ... G.LUP7、 G.LUPACT 以及 MTB.P。

1170 '#SYNC POS 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 只有标签的程序段号才能与该指令位于同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1171 '# 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 只有标签的程序段号可与 "#" 指令编写在同一程序段中。
唯一的例外为 #AXIS 指令，它必须与 G201 功能一块编写。
解决方案： 将它们编写在不同的程序段中。

1172 ' 半径补偿有效时，不允许使用该指令 '

检测时间： 执行程序时。
引起原因： 试图激活与刀具半径补偿不相容的功能。
解决方案： 取消刀具半径补偿，以激活该功能。

1173 '#UNLINK 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 只有标签的程序段号可与 "#" 指令编写在同一程序段中。
解决方案： 核对程序。



1174 '#LINK: 当前耦合 (随动) 有效时不能定义新的耦合 (随动) '

检测时间： 执行程序时。

引起原因： 在第一个耦合 （随动）未失效的情况下，试图激活第二个耦合 （随动）。

解决方案： 查阅编程手册，核对指令的语法。

- 如果只需要第二个耦合，在激活前先使用 #UNLINK 指令使第一个失效。
- 如果需要两个耦合，先使用#UNLINK指令使第一个失效，然后再使用一条#LINK 指令同时激活两个耦合。

1175 '#LINK: 未定义耦合 (随动)'

检测时间： 执行程序时。

引起原因： #LINK 指令未定义轴耦合 （随动）。

解决方案： 在 #LINK 指令中编写主轴和从轴。

1176 '#LINK: 主轴不属于当前的轴配置 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- #LINK 指令中编写的主轴在通道中不存在或不可用。
- 欲使用 #LINK 指令使耦合失效，该耦合的主轴在通道中不存在或不可用。

解决方案： 主轴与从轴必须存在于执行指令所在的通道中。

1177 '#LINK: 从轴不属于当前的轴配置 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- #LINK 指令中编写的主轴在通道中不存在或不可用。
- 欲使用 #LINK 指令使耦合失效，该耦合的主轴在通道中不存在或不可用。

解决方案： 主轴与从轴必须存在于执行指令所在的通道中。

1178 '#LINK: 从轴不能为主平面内的轴 '

检测时间： 执行程序时。

引起原因： 欲使用三根主轴里的一根作为从轴来激活耦合。

解决方案： 核对程序。

1179 '#LINK: 主轴与从轴的类型 (AXISTYPE) 必须一致 '

检测时间： 执行程序时。

引起原因： 欲激活一对耦合 (随动)，但主轴与从轴的机床参数 AXISTYPE （轴类型）不同。

解决方案： 主轴与从轴必须拥有相同的机床参数 AXISTYPE （轴类型）。

1180 '#LINK: 主轴与从轴的模式 (AXISMODE) 必须一致 '

检测时间： 执行程序时。

引起原因： 欲激活一对耦合 (随动)，但主轴与从轴的机床参数 AXISMODE （轴模式）不同。

解决方案： 主轴与从轴必须拥有相同的机床参数 AXISMODE （轴模式）。

1181 '#LINK:G201 中的有效轴不能定义为从轴 '

检测时间： 执行程序时。

引起原因： 欲将轴定义为耦合的从轴，该轴与 G201 功能一起使用，且处于附加手动模式。

解决方案： 如果需要耦合，可使用 G202 取消轴的附加手动模式。

1182 '#LINK: 编写的耦合过多 '

检测时间： 执行程序时。

引起原因： #LINK 指令激活的耦合数超出最大许可范围。

解决方案： #LINK 指令可激活的最大耦合数取决于系统的轴数量 (不计主轴)。上限为轴 -3。

1183 '#LINK 指令必须位于单独的程序段中 '

检测时间： 执行程序时。

引起原因： 只有标签的程序段号可与 "#" 指令编写在同一程序段中。

解决方案： 核对程序。

1184 '#LINK: 主轴与从轴相同 '

检测时间: 执行程序时。
引起原因: 欲激活耦合，其主轴与从轴相同。
解决方案: 主轴与从轴不能相同。

1185 '#LINK: 几根主轴的从轴不能为同一根轴 '

检测时间: 执行程序时。
引起原因: 将一根轴定义为几根主轴的从轴。
解决方案: 从轴只能有一根主轴。

1186 "#LINK: 主轴不能为另一耦合的从轴，反之亦然 '

检测时间: 执行程序时。
引起原因: 欲将轴定义为某耦合的主轴，同时将其定义为另一耦合的从轴。
解决方案: 轴不能同时为主轴和从轴。

1187 '#AXIS: 轴名重复 '

检测时间: 执行程序时。
引起原因: 同一轴在指令中定义了两次或两次以上。
解决方案: 核对程序。

1188 '"[" 指令多余 '

检测时间: 执行程序时。
引起原因: 指令语法错误。
解决方案: 查阅编程手册，检查指令语法。

1189 '#MPG: 参数过多 '

检测时间: 执行程序时。
引起原因: 指令编写的参数数量超出许可范围。
解决方案: #MPG指令最多可使用3个参数。它们分别代表每个手轮脉冲在开关每个位置上移动的距离。

1190 '#MPG: 手轮分辨率不能为负或 0'

检测时间: 执行程序时。
引起原因: 编写的手轮分辨率为负或 0。
解决方案: 每个手轮脉冲的运动距离必须为正且不能等于 0。

1191 '#INCJOG: 增量微动距离不能为负或 0'

检测时间: 执行程序时。
引起原因: 编写的增量微动距离为负或 0。
解决方案: 轴在开关每个位置上的增量微动值必须为正且不能等于 0。

1192 '#INCJOG: 增量微动进给率不能为负或 0'

检测时间: 执行程序时。
引起原因: 编写的增量微动距离为负或 0。
解决方案: 轴在开关每个位置上的进给率值必须为正且不能等于 0。

1193 '#CONTJOG/#INCJOG: 编写的进给率超出规定范围 '

检测时间: 执行程序时。
引起原因: 编写的轴进给率过大。
解决方案: 编写较小的值。

1194 '#INCJOG: 参数过多 '

检测时间: 执行程序时。
引起原因: 指令编写的参数数量超出许可范围。
解决方案: #INCJOG 指令最多可使用 5 组参数。它们分别代表在开关每个位置上的微动进给率和运动。

1195 '#CONTJOG: 参数过多 '

检测时间： 执行程序时。
引起原因： 指令编写的参数数量超出许可范围。
解决方案： #CONTJOG 指令只能使用一个参数，它表示开关位于连续微动模式时的轴进给率。

1196 '#CONTJOG: 连续微动进给率不能为负或 0'

检测时间： 执行程序时。
引起原因： 编写的连续微动进给率为负或 0。
解决方案： 轴进给率的值必须为正且不能为 0。

1197 '#SET OFFSET: 偏置下限为正 '

检测时间： 执行程序时。
引起原因： 编写的轴行程下限为正。
解决方案： 偏置下限值必须为负或 0。

1198 '#SET OFFSET: 负极限超出规定范围 '

检测时间： 执行程序时。
引起原因： 编写的轴极限值过低。
解决方案： 编写较大的值。

1199 '#SET OFFSET: 偏置上限为负 '

检测时间： 执行程序时。
引起原因： 编写的轴行程上限值为负。
解决方案： 上限值必须为正或 0。

1200 '#SET OFFSET: 正极限超出规定范围 '

检测时间： 执行程序时。
引起原因： 编写的轴极限值过高。
解决方案： 编写较小的值。

1201 '#SET OFFSET: 偏置上限与下限为 0'

检测时间： 执行程序时。
引起原因： 它表示轴的行程为 0。
解决方案： 核对程序。

1203 '#SET IPOPOS 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 只有标签的程序段号可与 "#" 指令编写在同一程序段中。
解决方案： 核对程序。

1204 ' 指令不存在或编程错误 '

检测时间： 执行程序时。
引起原因：
解决方案： 检查程序段的语法。

1205 '#CALL AX/#SET AX: 偏置类型未知 '

检测时间： 执行程序时。
引起原因： 指令中编写的偏置类型不存在。
解决方案： 有效的偏置类型有：ALL、LOCOF、FIXOF、TOOLOF、ORGOF、MEASOF、MANOF。

1206 ' 需要 ",,"'

检测时间： 执行程序时。
引起原因： 编写的指令或功能中需要 ","。
解决方案： 检查程序段语法。



1209　' 轴索引超出规定范围 '

检测时间：　执行程序时。

引起原因：　可能的原因包括：

1. #CALL AX/#SET AX：轴的位置不正确。有可能是索引超出最大范围，也有可能是在未指示位置时，轴与上一轴之间的没有足够空间。
2. 轴名中使用了错误的字符。

解决方案：　解决的办法有：

1. #CALL AX/#SET AX：可将轴置于任何未占用的位置，该位置介于 1 ～系统许可的最大轴数量 + 最大主轴数之间。
2. 无效字符可能为 @1 ～ @6 以及 @SM。

1210　'#CALL AX/#SET AX: 轴名重复 '

检测时间：　执行程序时。

引起原因：　同一轴编进行了两次编程。

解决方案：　检查轴名，同一轴不能在通道中重复。

1211　'#CALL AX/#SET AX: 轴索引重复 '

检测时间：　执行程序时。

引起原因：　欲在同一位置放置 2 根轴。

解决方案：　检查通道需要的轴配置，两根轴不能位于同一位置。

1213　'#CALL AX/#SET AX: G63 有效时不允许使用 '

检测时间：　执行程序时。

引起原因：　G63 功能有效时不能编写 #CALL AX。

解决方案：　在更改轴配置前先让 G63 加工螺纹失效。

1214　'#CALL AX/#SET AX: 请求的轴过多 '

检测时间：　执行程序时。

引起原因：　可能是请求的轴过多，也有可能是将它们放在了大于通道最大值的较高位置上。

解决方案：　通道中使用的轴不能超出系统许可的最大轴数量，轴的有效位置介于 1 ～系统许可的最大轴数量 + 最大主轴数之间。

1215　'#CALL AX/#SET AX 指令必须位于单独的程序段中 '

检测时间：　执行程序时。

引起原因：　只有标签的程序段号可与 "#" 指令编写在同一程序段中。

解决方案：　核对程序。

1216　'#CALL AX/#CAX: 轴名已经使用 '

检测时间：　执行程序时。

引起原因：　指令中为作为 C 轴使用的主轴编写的名字已经被通道中的另一根轴使用。

解决方案：

1217　'#CALL AX: 索引已经使用 '

检测时间：　执行程序时。

引起原因：　#CALL AX 中指示的轴位置被另一根轴占用。

解决方案：　检查通道所需的轴配置，两根轴不能位于同一位置。其中一根轴可以置于任何未被占用的，介于 1 ～系统许可的最大轴数量 + 最大主轴数之间的位置。

1218　'#FREE AX 指令必须位于单独的程序段中 '

检测时间：　执行程序时。

引起原因：　只有标签的程序段号可与 "#" 指令编写在同一程序段中。

解决方案：　查阅手册，检查指令的语法。

1219　' 需要 "," 或 "]"'

检测时间：　执行程序时。

引起原因：　指令语法错误。

解决方案：　查阅编程手册，检查指令语法。

1220 '#FREE AX: 手动模式中不能删除活动轴 '

检测时间： 执行程序时。
引起原因： 不能从通道中删除有效功能 G201 (附加手动) 中的轴。
解决方案： 要删除这些轴，首先取消 G201，或者在激活该功能前，别在它后面编写相应的轴。

1221 '#SET AX 指令必须位于单独的程序段中 '

检测时间： 执行程序时。
引起原因： 只有标签的程序段号可与 "#" 指令编写在同一程序段中。
解决方案： 查阅手册，检查指令语法。

1222 '#COMMENT END 指令多余 '

检测时间： 执行程序时。
引起原因： 未在 #COMMENT END 前编写 #COMMENT BEGIN。
解决方案： 核对程序。

1223 ' 指令程序段中需要 End-of-file （文件结束）字符 '

检测时间： 执行程序时。
引起原因： #COMMENT BEGIN 程序段没有对应的 #COMMENT END。
解决方案： 确定程序中每个 #COMMENT BEGIN 程序段均有对应的 #COMMENT END。

1224 ' 算子未知或缺失 '

检测时间： 执行程序时。
引起原因： 在写变量或参数的指令中，算子未知或缺失。
解决方案： 有效分配算子为 "="、 "+="、 "-="、 "*="、 "/="。

1225 ' 除数为 0'

检测时间： 执行程序时。
引起原因： 编写的运算操作中，除数为 0。
解决方案： 除数只能为 0 以外的数。在编程历史记录中，使用的参数有可能取了 0 值。确保作为除数使用的参数值不为 0。

1226 ' 有效平面内缺失用作半径补偿的曲面轴 (FACEAXIS) '

检测时间： 执行程序时。
引起原因： 在机床参数中，必须将有效平面内的一根轴定义为 FACEAXIS 类型。
解决方案： 在机床参数中，将其中一根轴定义为 FACEAXIS。

1227 ' 有效平面内缺失用作半径补偿的纵轴 (LONGAXIS) '

检测时间： 执行程序时。
引起原因： 在机床参数中，必须将有效平面内的一根轴定义为 LONGAXIS 类型。
解决方案： 在机床参数中，将其中一根轴定义为 LONGAXIS。

1233 ' 零点偏置超出数值范围 '

检测时间： 执行程序时。
引起原因： 零点偏置的值大于设定的最大值。
解决方案： 核对程序。

1236 ' 宏名过长 '

检测时间： 执行程序时。
引起原因： 宏名字符数超出最大许可值。
解决方案： 最多为 30 个字符。

1237 ' 与宏有关的文本需要 "\"'

检测时间： 执行程序时。
引起原因： 在宏替换文本中，文本位于引号之间，但引号前并无 "\" 字符。
解决方案： 在宏替换文本中，引号前必须有 "\" 字符。连接宏的一个明显实例是 "macro" = " \"macro1\" \"macro2\" "。



1238 ' 宏的替换文本过长 '

检测时间: 执行程序时。
引起原因: 宏替换文本的字符数超出最大许可值。
解决方案: 最大为 140 个字符。

1239 ' 宏过多 '

检测时间: 执行程序时。
引起原因: 超出程序中可定义的宏最大数量。
解决方案: 最多为 50 个宏。

1240 ' 轴不存在 '

检测时间: 执行程序时。
引起原因: 欲使用前面未定义的宏。
解决方案: 定义宏为 "macro" = " 替换文本 "。
查阅编程手册，检查宏语法。

1241 ' 宏的替换文本缺失 '

检测时间: 执行程序时。
引起原因: 宏赋值为空字符串或字符。
解决方案: 根据宏的功能为其赋予正确的替换文本。在宏定义中，替换文本必须位于两个引号之间。

1244 ' 曲面轴靠近旋转中心：G96 的主轴转速受限 '

检测时间: 执行程序时。
引起原因: 在恒表面速度下工作时，主轴转速受限。曲面轴太靠近旋转中心导致转速受限。
解决方案: 别让轴太靠近旋转中心，可增加最大许可转速或接受该极限。

1245 'G96: 有效平面内未定义曲面轴 '

检测时间: 执行程序时。
引起原因: 有效平面内未定义任何曲面轴。
解决方案: 设置曲面轴的机床参数：FACEAXIS = Yes。

1246 ' 不能使用 G95 中的进给率加工螺纹 '

检测时间: 执行程序时。
引起原因: 欲在 G95 功能有效时 (进给率单位为毫米 / 转或英寸 / 转) 执行刚性攻丝 (G63)。
解决方案: 使用 G94 功能编写进给率 (毫米 / 分钟或英寸 / 分钟)。

1247 'G96 有效时不能加工螺纹 '

检测时间: 执行程序时。
引起原因: 欲在 G96(恒表面速度) 功能有效时执行刚性攻丝 (G63)。
解决方案: 利用 G97 功能取消 G96 功能。

1248 ' 攻丝与 G192 不能位于同一程序段中 '

检测时间: 执行程序时。
引起原因: G63 (刚性攻丝) 与 G192 (恒表面速度限制) 不能编写在同一程序段中。
解决方案: 将它们编写在不同的程序段中。

1249 ' 不能在 G96 有效时换档 '

检测时间: 执行程序时。
引起原因: 欲在 G96 功能有效时使用 G112 功能进行主轴换档。
解决方案: 取消 G96 功能，进行主轴换档。

1251 'G96 有效时不能使用手动模式 '

检测时间: 执行程序时。
引起原因: G96 功能有效时不能将开关切换至手动模式。
解决方案: 取消 G96 功能，切换至手动模式。

1252 '#FREE AX: G96 有效时不能删除端面车削轴 '

检测时间： 执行程序时。

引起原因： 使用#FREE AX 指令且恒表面速度有效时，不能从通道中删除曲面轴(FACEAXIS) 。

解决方案： 使用 G97 功能取消恒表面速度，然后删除轴。

1254 'G192 与 M19 不能位于同一程序段中 '

检测时间： 执行程序时。

引起原因： M19 (主轴定位) 与 G192 (恒表面速度限制) 不能位于同一程序段中。

解决方案： 将它们编写在不同的程序段中。

1255 ' 加速度百分比为负 '

检测时间： 执行程序时。

引起原因： 使用 G130 功能时，使用负的加速度百分比。

解决方案： 加速度百分比必须等于或大于 0。

1256 ' 加速度百分比超出规定范围 '

检测时间： 执行程序时。

引起原因： G130 功能编写的加速度百分比过大。

解决方案： 加速度百分比最大许可值为 2147483646。

1257 ' 丝杠螺距重复编程 '

检测时间： 执行程序时。

引起原因： 使用 G33 功能时，程序段中编写了两次或两次以上螺距。

解决方案： 在程序段中仅定义一次螺距。

1258 ' 丝杠螺距等于 0'

检测时间： 执行程序时。

引起原因： G33 功能编写的螺距为 0。

解决方案： 使用与 X、 Y、 Z 轴分别对应的字母 I、 J、 K 编写螺距。
查阅编程手册，检查指令语法。

1259 ' 丝杠螺距超出规定范围 '

检测时间： 执行程序时。

引起原因： G33 功能编写的螺距过大。

解决方案： 编写较小的螺距值。

1261 ' 运动类型未知 '

检测时间： 执行程序时。

引起原因： 无有效运动时，欲在模式 6 中激活 #RTCP、或 #TLC、或 #CS/ACS。

解决方案： 首先激活运动，然后是转换。
#KIN ID [ ]
#RTCP ON 或 #TLC ON [ ] 或 #CS/ACS ON [MODE6, , , ,]

1262 ' 转换所需的轴组不够 '

检测时间： 执行程序时。

引起原因： 坐标转换 #RTCP、或 #TLC、或 #CS/ACS 需要许多有效轴，而且它们必须为通道的第一轴。根据有效运动的类型，这些有效轴必须介于 3 ～ 5 之间。

解决方案： 使用 #SET AX [, , , ,] 激活轴，并在通道的第一位置应用转换。

1263 ' 转换的旋转轴缺失 '

检测时间： 执行程序时。

引起原因： 编写了 #TOOL ORI 指令，但是没有用来放置与所定义斜面正交的刀具的旋转轴。

解决方案： 不使用 #TOOL ORI 指令，或者激活运动，以允许放置与所定义斜面正交的刀具。

1264 'CS/ACS 有效时不允许编程 '

检测时间: 执行程序时。

引起原因: CS 或 ACS 有效时试图执行以下功能。

- 原点搜索 (G74)。
- 修正软件极限 (G198 - G199)。
- 执行探测循环 #PROBE1。
- #OSC 指令。
- #LINK 指令。
- #ECS 指令。

解决方案: 首先取消有效斜面，然后执行欲使用的指令。最后再重新激活斜面。

1265 'RTCP/TLC 有效时不允许编程 '

检测时间: 执行程序时。

引起原因: RTCP 或 TLC 有效时试图执行以下功能。

- 原点搜索 (G74)。
- 修正软件极限 (G198 - G199)。
- #OSC 指令。
- #KIN ID 指令。

解决方案: 首先取消有效斜面，然后执行欲使用的指令，最后再重新激活斜面。

1266 ' 由于使用 #TLC OFF 指令使 TLC 功能无效 '

检测时间: 执行程序时。

引起原因: #TLC 有效时，使用了 #TLC 指令而不是 #TLC OFF。

解决方案: 要重新编写 #TLC ON 以改变前面的编程值，必须先使用 #TLC OFF 进行取消。

1268 '#CS ON/#ACS ON: 语法错误 '

检测时间: 执行程序时。

引起原因: 指令语法错误。

解决方案: 查阅编程手册，检查指令语法。

1269 '#CS ON/#ACS ON: 编程角度无效 '

检测时间: 执行程序时。

引起原因: 编写的角度无效。

解决方案: 指令中编写的角度必须在 ± 360º 以内。

1270 ' 不能计算坐标转换 '

检测时间: 执行程序时。

引起原因: 试图进行从工件到机床或从机床到工件的坐标转换，未能实现。

解决方案: 取消转换，改变目标位置，然后重新激活转换。

1271 ' 有效运动的轴不能被排除或更改 '

检测时间: 执行程序时。

引起原因: 欲使用 #CALL AX、 #SET AX、 #FREE AX 指令改变影响有效运动的轴。

解决方案: 在改变通道轴之前取消运动。#KIN ID [0]。

1272 ' 有效转换的轴不能被排除或更改 '

检测时间: 执行程序时。

引起原因: 在有效斜面中，欲使用 #CALL AX[ ]、 #SET AX[ ]、 #FREE AX[ ] 指令改变影响转换的第一个三根轴里的一根。

解决方案: 在改变通道轴之前取消斜面转换。#CS/ACS OFF ALL。

1277 ' 计算出的零点偏置超出数值范围 '

检测时间: 执行程序时。

引起原因: 根据 G92 功能编写的坐标计算出的零点偏置过大。

解决方案: 核对程序。

1278 'G131/G133: 值错误'

检测时间: 执行程序时。
引起原因: 可能的原因包括:
- G131 功能编写的轴加速度百分比值错误。
- G133 功能编写的轴加加速度百分比值错误。

解决方案: 轴的加速度或加加速度百分比必须为正且小于或等于 100%。

1279 ' 需要 "'

检测时间: 执行程序时。
引起原因: 表达式或指令中缺失引号。
解决方案: 检查程序段的语法。

1281 " 参数个数与格式指示符不匹配 '

检测时间: 执行程序时。
引起原因: #MSG或#WARNING 指令中出现的格式指示符 (%D或%d)个数与消息中显示的参数个数不匹配。
解决方案: 查阅编程手册，检查指令语法。

1282 ' 消息过长 '

检测时间: 执行程序时。
引起原因: #MSG、 #ERROR、 #WARNING 指令中编写的消息超过 69 个字符。
解决方案: 缩减消息中的字符数，记住最大字符数为 69。该最大值包括替换指示符的字符。

1283 ' 格式指示符过多 '

检测时间: 执行程序时。
引起原因: #MSG、#ERROR、#WARNING 指令中编写的格式指示符 (%D 或 %d) 多于 5 个。
解决方案: 减少格式指示符数量。

1284 ' 需要算术表达式 '

检测时间: 执行程序时。
引起原因: #MSG、#WARNING、#ERROR 指令中编写了格式指示符 (%D 或 %d)，但是未编写需要显示的值清单。
解决方案: 在指令中编写需要显示的值清单。
查阅编程手册，检查指令语法。

1285 ' 刀具半径重复编写 '

检测时间: 执行程序时。
引起原因: 同一程序段中编写了两次写刀具半径。
解决方案: 在程序段中只编写一次写刀具半径。

1286 ' 刀具长度重复编写 '

检测时间: 执行程序时。
引起原因: 同一程序段中编写了两次写刀具长度。
解决方案: 在程序段中只编写一次写刀具长度。

1287 ' 需要 "["'

检测时间: 执行程序时。
引起原因: 表达式或指令中缺失左括弧。
解决方案: 检查程序段语法。

1288 ' 指令中编写的参数过多 '

检测时间: 执行程序时。
引起原因: 指令编程错误。
解决方案: 查阅编程手册，检查指令语法。

1290 'I、 J、 K 坐标编程错误 '

检测时间： 执行程序时。

引起原因： 可能的原因有：

- 圆弧插补圆心、坐标系统的极点或旋转中心的编程值过大。
- G264 有效时，圆弧插补的圆心编程坐标值错误。

解决方案： 核对程序。

1291 ' 不允许再使用 S 功能 '

检测时间： 执行程序时。

引起原因： 同一程序段中使用的 S 功能个数超出最大许可值。

解决方案： 同一程序段中允许使用的 S 功能最多为 4 个。

1292 'M 功能重复编程 '

检测时间： 执行程序时。

引起原因： 与主轴不相关的 M 功能在同一程序段中编写了两次或两次以上。

解决方案： 将它们编写在不同的程序段中。

1293 'H 功能重复编程 '

检测时间： 执行程序时。

引起原因： H 功能在同一程序段中编写了两次或两次以上。

解决方案： 将它们编写在不同的程序段中。

1301 ' 刀具长度转换超出有效数字格式 '

检测时间： 执行程序时。

引起原因： 超出了刀具长度转换中的最大数字格式。

解决方案： 修正转换的值或刀具值。

1302 ' 名字中存在错误字符 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

1. 标签名中有错误字符。
2. 子程序名中有错误字符。
3. 外部变量名中有错误字符。

解决方案： 以上各种情形的有效字符分别为：

1. 除了引号、"]" 以及 TAB 以外的所有字符。
2. 除了引号、空格以及 TAB 以外的所有字符。 "(" 后的所有字符均为注释。
3. 字母、数字以及 "_" 符号。

1303 ' 变量名过长 '

检测时间： 执行程序时。

引起原因： 变量名中，将 "工件 " 定义为一组由 "." 分隔的字符，当工件多于 13 个字符时便出错。

解决方案： 缩短变量名。

1304 ' 主轴转速错误 '

检测时间： 执行程序时。

引起原因： 主轴转速编程值过低。

解决方案： 主轴转速值必须大于 1.6 10E-7 rpm。



1305 '#MCS 有效时不允许进行编程 '

检测时间： 执行程序时。

引起原因： #MCS 有效时编写了下面的一种功能：

- 零点偏置 (G54-G59、 G159、 G92、 G158、 G101、 G102、 G53) 开 / 关。
- 夹具开 / 关 ("V.G.FIX" 变量)。
- 镜像 (G11/G12/G13/G14) 开 / 关。
- 半径 / 直径编程 (G151/G152)。
- 激活增量编程 (G91)。
- 毫米 / 英寸编程 (G70/G71)。
- 比例缩放因子 (G72)。
- 极坐标中的 G0、 G1、 G2、 G3、 G8 或 G9 运动。
- 极坐标中的 G63 或 G33 加工螺纹。
- 极点 (G30)。
- 模式旋转 (G73)。
- #FACE、 #CYL 与 #RTCP 指令。

解决方案： 核对程序。

1306 ' 刀具半径补偿有效时不能改变运动 '

检测时间： 执行程序时。

引起原因： 试图在刀具半径补偿 (G41/G42) 有效时改变有效运动。

解决方案： 在改变有效运动前先取消刀具半径补偿。

1308 ' 有效转换的轴不能为从轴 '

检测时间： 执行程序时。

引起原因： 使用 #LINK 指令将有效运动转换中的轴定义为耦合的从轴。

解决方案： 取消运动，激活以该轴为从轴的耦合。
有效运动中的轴可以为耦合的主轴。

1309 ' 需要文件名 '

检测时间： 执行程序时。

引起原因： 未选择执行的文件。

解决方案： 选择欲执行的文件。

1310 ' 程序线过长 '

检测时间： 执行程序时。

引起原因： #EXBLK 指令中编写的程序段的字符数超出最大许可范围。

解决方案： 使用 #EXBLK 指令编写的程序段的最大许可字符数为 128。

1311 ' 编程轴中未包括测量偏置 '

检测时间： 执行程序时。

引起原因： 欲在未包括测量偏置 (G101) 的轴上删除测量偏置 (G102) 。

解决方案： 如果前面未执行 G101 功能，那么执行 G102 功能是没有意义的。

1314 '#CS ON/#ACS ON: 标识符错误 '

检测时间： 执行程序时。

引起原因： 指令中编写的 CS/ACS 类标识符值错误。

解决方案： CS/ACS 类标识符必须为正值、不等于 0 且小于或等于 5。

1315 '#CS ON/#ACS ON: 系统未定义 '

检测时间： 执行程序时。

引起原因： 编写的 #CS ON/#ACS ON 指令无参数，而前面并未定义或激活转换。

解决方案： 编写无参数的 #CS ON/#ACS ON 指令时，CNC 试图激活最近存储的转换。在这种情况下，由于并未存储转换，因而它是不可能实现的。
为 #CS/#ACS 编写适当的参数，以激活需要的转换。
查阅编程手册，检查指令语法。



1316 '#CS/#ACS DEF: 参数缺失 '

检测时间: 执行程序时。
引起原因: 指令所需参数未编程。
解决方案: #CS/#ACS DEF 指令需要编写模式、转换向量以及旋转角度。
查阅编程手册，检查指令语法。

1318 '#CS ON/#ACS ON: 不允许改变有效坐标系统 '

检测时间: 执行程序时。
引起原因: 试图改变有效转换的参数。
解决方案: 不能使用 #CS/#ACS DEF 指令更改早先定义以及未激活的转换的参数。
可以更改已经定义但尚未激活的参数。

1319 '#CS ON/#ACS ON 指令嵌套超出规定范围 '

检测时间: 执行程序时。
引起原因: 超出最大可交迭转换数目。
解决方案: 可交迭的 #CS/#ACS 转换最高为 10 个。

1320 ' 标签过多 '

检测时间: 当前版本不会出现该问题。核对程序。
引起原因: 程序中定义的 "Nxxx:" 或 " [xxx]" 类标签大于 128 个。
解决方案: 删除标签，直到每种类型的标签数低于 128 个。

1321 ' 标签名过长 '

检测时间: 执行程序时。
引起原因: 标签名超过 15 个字符。
解决方案: 减少标签名的字符数。

1322 ' 标签重复定义 '

检测时间: 执行程序时。
引起原因: 在程序的不同地方重复定义标签。
解决方案: 删除重复定义的标签。

1323 '$GOTO: 标签错误 '

检测时间: 执行程序时。
引起原因: 标签只能使用位于括弧之间的字符串或 "N" 字符后跟小于 2147483646 的正数来定义。
解决方案: 删除标签，直到每种类型的标签数均小于 128 个。

1324 ' 标签未定义 '

检测时间: 执行程序时。
引起原因: $GOTO 或 #RPT 指令中编写了未定义的标签。
解决方案: 在某些程序点上定义跳转标签。

1325 ' 程序段号重复定义 '

检测时间: 执行程序时。
引起原因: 在程序的不同地方重复定义同一程序段号 "N"。
解决方案: 不要重复定义程序段号。

1326 ' 变量赋值错误 '

检测时间: 执行程序时。
引起原因: 为变量赋的值过大。
解决方案: 核对程序。

1327 ' 主轴定位转速重复编程 '

检测时间: 执行程序时。
引起原因: 程序段中重复编写主轴定位转速 (M19) 。
解决方案: 在程序段中仅编写一次定位转速。



1328 '$FOR 指令没有对应的 $ENDFOR 指令 '

检测时间： 执行程序时。
引起原因： 程序段中编写了 $FOR 指令，但是后面并没有 编写 $ENDFOR 指令。
解决方案： 核对程序。

1330 ' 镜像编程错误 '

检测时间： 执行程序时。
引起原因： G14 功能 (镜像) 编程错误。
解决方案： G14 功能可激活 (因子 -1) 或取消 (因子 1) 轴的镜像。

1331 '#TANGFEED RMIN: 不允许使用负半径 '

检测时间： 执行程序时。
引起原因： #TANGFEED RMIN 的半径必须大于 0。
解决方案： 核对程序。

1332 '#TOOL AX: 指定轴后还需要定位 +/-'

检测时间： 执行程序时。
引起原因： 未编写刀具定位。
解决方案： 查阅编程手册，检查指令语法。

1333 ' 在刀具半径补偿有效时更改平面的第一和 / 或第二轴 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 刀具半径补偿有效时，试图使用#CALL AX或#SET AX指令更改平面的第一或第二轴。
- 刀具半径补偿有效时，试图使用 #FREE AX 指令释放平面的第一或第二轴。

解决方案： 取消刀具半径补偿，再对平面的第一或第二轴进行操作。

1334 'G200: 不允许在同一程序段中编写运动 '

检测时间： 执行程序时。
引起原因： 在 G200 功能行中编写了轴运动。
解决方案： 编写在不同行中。

1336 ' 配置错误 : 两根 CAXIS 轴 '

检测时间： 执行程序时。
引起原因： #FACE/#CYL 指令中的两根轴均为 C 轴。
解决方案： 两根轴中只能有一根 C 轴。换句话说，令机床参数 CAXIS = Yes。

1337 ' 未定义 CAXIS'

检测时间： 执行程序时。
引起原因： #FACE/#CYL 指令中未定义 C 轴。
解决方案： 该指令中的两根轴中必须有一根 C 轴。换句话说，令机床参数 CAXIS = Yes。

1339 ' 选择无效 '

检测时间： 执行程序时。
引起原因： 编程指令无效，因为它跟前面的指令类似，因而它本身已经是有效的。
解决方案： 核对程序。

1340 ' 取消选择无效 '

检测时间： 执行程序时。
引起原因： 编写了 #CAX OFF 指令，但是主轴并非 C 轴。换句话说，前面并未编写 #CAX 指令。
解决方案： 核对程序。

1342 ' 转换有效时不允许使用 #CAX OFF'

检测时间： 执行程序时。
引起原因： #RTCP 或 #TLC 有效时 C 轴不能失效。
解决方案： 核对程序。

1343 ' 运动类型有效时不允许使用 #FACE OFF'

检测时间： 执行程序时。
引起原因： #FACE OFF 指令前没有使用 #FACE 指令激活工件表面的加工操作。
解决方案： 核对程序。

1344 ' 加工工件侧面时，不允许改变平面 '

检测时间： 执行程序时。
引起原因： 侧面加工有效时编写了 G17-G20 功能。
解决方案： 核对程序。

1345 'G20: 轴编程错误 '

检测时间： 执行程序时。
引起原因： 使用 G20 功能 (改变平面) 时，平面的前两根轴 (参数 1 和 2) 编程错误。
解决方案： 这两根轴必须不同且它们必须为配置的主轴。

1347 ' 运动类型有效时不允许使用 #CYL OFF'

检测时间： 执行程序时。
引起原因： #CYL OFF 指令前没有使用 #CYL 指令激活工件侧面的加工操作。
解决方案： 核对程序。

1348 '#CYL: 半径错误 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
1. #CYL 指令中的半径为负或 0。
2. 如果半径为变量，穿越正在生成的圆柱体的圆心，从而得到 0 半径。

解决方案： 解决的办法有：
1. 在 #CYL 指令中编写非 0 的正半径 。
2. 不要穿越正在生成的圆柱体的圆心。

1349 ' 激活 #FACE 时轴坐标为负 '

检测时间： 执行程序时。
引起原因： 直线轴为平面轴 C 转换的一部分，它位于旋转轴的负向。
解决方案： 可采用的解决办法有：
- 激活平面轴 C 时，将轴置于旋转中心的正向。
- 定义平面轴C的类型，可通过改变通道的通用参数ALINGC=NO使轴通过旋转中心。

1350 '#VAR/#ENDVAR 指令间字符错误 '

检测时间： 执行程序时。
引起原因： 指令间的某个程序段中存在错误字符。
解决方案： 这些指令之间只能存在用户变量声明 (如果一行有几个，以逗号分隔) 或程序段号 。

1351 '#VAR/#ENDVAR/#DELETE: 变量类型不允许 '

检测时间： 执行程序时。
引起原因： 试图定义或删除不是用户变量的变量， V. P 或 V. S。
解决方案： 仅定义或删除用户变量。

1352 '#VAR/#ENDVAR: 定义的变量已经存在 '

检测时间： 执行程序时。
引起原因： 定义的用户变量已经在前面定义过。
解决方案： 核对程序。

1353 ' 初始化数组的值过多 '

检测时间： 执行程序时。
引起原因： 初始化用户数组变量时，初始化的位置多于数组本身的数量。
解决方案： 核对程序。

1354 ' 读变量出错 '

检测时间： 执行程序时。
引起原因： 读变量时发生错误。
解决方案： 核对程序。

1355 ' 不能删除变量 '

检测时间： 执行程序时。
引起原因： 试图删除 CNC 变量。
解决方案： 只能删除用户定义的变量 (前缀为 P 和 S)。

1356 ' 需要变量或参数 '

检测时间： 执行程序时。
引起原因： $IF EXIST 指令编程错误。
解决方案： $IF EXIST 控制指令只能使用算术参数或变量。

1357 '#DELETE: 字符错误 '

检测时间： 执行程序时。
引起原因： 该指令所在程序段存在错误字符。
解决方案： 该指令必须位于单独的程序段中，或者紧挨程序段号。
只能删除前面定义的用户变量 V. P 或 V. S (如果同一行中存在多个变量，则以逗号分隔)。

1358 '#DELETE: 欲删除的变量不存在 '

检测时间： 执行程序时。
引起原因： 试图删除已经被删除的用户变量。
解决方案： 核对程序。

1360 'C 轴有效时，不允许使用 G33/G63/G95/G96/G97'

检测时间： 执行程序时。
引起原因： C 轴有效时，编写了 G33/G63/G95/G96/G97 功能中的一种。
解决方案： 可使用 #CAX OFF 指令让 C 轴失效。

1363 ' 数组变量声明错误 '

检测时间： 执行程序时。
引起原因： 用户变量 (V.P、 V.S) 的声明必须位于 #VAR/#ENDVAR 指令之间。
解决方案： 核对程序。

1364 ' 数组变量索引过多 '

检测时间： 执行程序时。
引起原因： 定义的多维数组用户变量多于 4 维。
解决方案： 核对程序。

1365 ' 主轴转速不允许为负 '

检测时间： 执行程序时。
引起原因： 程序段中编写的主轴转速为负。
解决方案： 主轴转速必须为正， G63 功能除外。

1367 ' 换档与主轴运动不能同时进行 '

检测时间： 执行程序时。
引起原因： 同一程序段中编写了主轴运动 M 功能与 G112 功能 (参数设置转换)。
解决方案： 将它们编写在不同的程序段中。

1368 ' 不能同时编写圆心与半径 '

检测时间： 执行程序时。
引起原因： 编写了圆弧插补的圆心与半径。
解决方案： 圆弧插补中，必须编写终点与半径或圆心的坐标。



1369 '#HSC: 不允许编程 '

检测时间： 执行程序时。
引起原因： 指令语法错误。
解决方案： 查阅编程手册，检查指令语法。

1370 '#HSC: 重复编程 '

检测时间： 执行程序时。
引起原因： 程序段中编写了两次 HSC 激活或失效指令。
解决方案： 程序段中只能编写 #HSC ON 或 #HSC OFF。

1374 ' 需要 M02/M30'

检测时间： 执行程序时。
引起原因： 主程序末尾没有编写 M02 或 M30 指令。
解决方案： 核对程序。

1375 ' 需要 M17/M29/#RET'

检测时间： 执行程序时。
引起原因： 子程序末尾没有编写 M17、 M29 或 #RET 指令。
解决方案： 核对程序。

1376 'C 轴未定义缺省名 '

检测时间： 执行程序时。
引起原因： #CAX指令未带参数，以C轴工作的通道主轴名由机床参数CAXIS指定。当机床参数未指定任何名字时出错。
解决方案： 主轴作为 C 轴使用时，在机床参数 CAXIS 中指定主轴名。

1377 ' 参数索引错误 '

检测时间： 执行程序时。
引起原因： 写入不存在的算术参数或该参数写保护。
解决方案： CNC 中可用的算术参数号由下列机床参数设置：
- MINLOCP-MAXLOCP：局部参数。
- MINGLBP-MAXGLBP：全局参数。
- MINCOMP-MAXCOMP：通用参数。

机床参数 ROPARMIN - ROPARMAX 定义的全局参数写保护。

1378 ' 读参数时索引错误 '

检测时间： 执行程序时。
引起原因： 读不存在的算术参数。
解决方案： CNC 中可用的算术参数号由下列机床参数设定：
- MINLOCP-MAXLOCP：局部参数。
- MINGLBP-MAXGLBP：全局参数。
- MINCOMP-MAXCOMP：通用参数。

1380 ' 固定循环编程错误 '

检测时间： 执行程序时。
引起原因： 程序段中，固定循环后不能编写任何内容。
解决方案： 将它们编写在不同的程序段中。

1381 ' 固定循环不存在 '

检测时间： 执行程序时。
引起原因： 编写的固定循环不存在。
解决方案： 参考编程手册中的固定循环。

1382 ' 固定循环中不允许使用该参数 '

检测时间： 执行程序时。
引起原因： 编写的参数不允许在固定循环中使用。
解决方案： 查阅编程手册，核对循环的语法。

1383 ' 固定循环中未编写应有的参数 '

检测时间： 执行程序时。
引起原因： 未编写固定循环所需的参数。
解决方案： 查阅编程手册，核对循环的语法。

1384 'M 功能不能与运动一块使用 '

检测时间： 执行程序时。
引起原因： 运动与带关联子程序并在运动前执行的 M 功能编写在同一程序段中 。子程序通常在程序段末尾执行，因此 M 功能决不会在运动前执行。
解决方案： 可采用的解决办法有：
- 不为 M 功能编写关联子程序。
- 将 M 功能设置在机床参数表中，在运动后执行。

1385 ' 不能在同一程序段中更改 D 与刀具长度 '

检测时间： 执行程序时。
引起原因： 试图在编写换刀与或改变刀具偏置的程序段中写 "V.G.TOL" 变量。
解决方案： 编写在不同行中。

1386 ' 不能在同一程序段中更改 D 与刀具偏置 '

检测时间： 执行程序时。
引起原因： 试图在编写换刀与或改变刀具偏置的程序段中写 "V.A.TOFL.axis" 变量。
解决方案： 核对程序。

1387 ' 程序段中 M 功能过多 '

检测时间： 执行程序时。
引起原因： 超出同一程序段中允许编写的最大 M 功能个数。
解决方案： 同一程序段中允许使用的最大 M 功能个数取决于软件版本：
- V1.10 以前的版本最多可使用 7 个 M 功能。
- 从 V2.00 版本开始，最多可使用 14 个 M 功能。

1388 ' 不允许再使用 H 功能 '

检测时间： 执行程序时。
引起原因： 超出同一程序段中允许使用的最大 H 功能个数。最多为 7 个。
解决方案： 将它们编写在不同的程序段中。

1389 'G 功能 (G10/G11/G12/G13/G14) 不相容 '

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1390 'G 功能 (G98/G99) 不相容 '

检测时间： 执行程序时。
引起原因： 同组的两个或多个 G 功能不能编写在同一程序段中。
解决方案： 将它们编写在不同的程序段中。

1392 ' 参数重复编程 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
- 某个参数在 #PCALL 调用、G180-G189 或固定循环中编写了两次。
- 某个参数在 #POLY 指令中编写了一次以上。

解决方案： 删除重复的参数。

1393 ' 当前刀具和刀具偏置与先前编写的不匹配 '

检测时间： 在程序段搜索模式中。
引起原因： 主轴上带偏置 D 的刀具与到达停止点前编写的带偏置 D 的刀具不匹配。 CNC 中有重复运动程序段，在利用编程刀具半径进行重定位后，使用该程序段来加工工件。如果主轴上的刀具不同且程序中对刀具半径进行补偿，那么将加工出不同的工件。
解决方案： 更改主轴上的刀具，以与先前编程的刀具匹配。



1394 ' 与 G 功能关联的子程序不存在 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- G74 功能位于单独程序段中，但是机床参数 REFPSUB 无关联子程序。
- G180-G189 功能中的一个位于单独程序段中，但是机床参数 OEMSUB 无关联子程序。

解决方案： 在对应的参数中定义欲执行和生效的子程序名。

1395 ' 带关联子程序的 G74 功能必须位于单独程序段中 '

检测时间： 执行程序时。

引起原因： G74 功能所在程序段中只能编写程序段号与标签，以执行关联子程序。

解决方案： 将它们编写在不同的程序段中。

1396 'MDI 模式中不允许编程 '

检测时间： 执行程序时。

引起原因： 该指令不能在 MDI 模式中执行。

解决方案： 在程序中执行该指令。

1397 'Hirth 轴编程位置错误 '

检测时间： 执行程序时。

引起原因： Hirth 轴的编程坐标未对应一整步。

解决方案： 编程坐标为 REFVALUE ＋ n × HPITCH。

1398 ' 参数 HIRTH = NO 的轴不能激活为 Hirth 轴 '

检测时间： 执行程序时。

引起原因： 在机床参数 HIRTH = NO 时试图激活 Hirth 轴或使其失效 (G171 与 G170)。

解决方案： 设置轴的机床参数：HIRTH = Yes。

1399 ' 不能将该轴激活为 Hirth 轴 '

检测时间： 执行程序时。

引起原因： 试图在以下几种情形时激活 Hirth 轴：

- 该轴为平面的第一或第二轴，且刀具半径补偿有效。
- 该轴为平面的第一或第二轴，且冲突检测有效。
- 该轴为第一个三根轴中的一根且 #CS 或 #ACS 转换有效。
- 该轴为第一个五根轴中的一根且5-轴转换 (#RTCP、#TLC或#TOOL ORI) 有效 。

解决方案： 核对程序。

1400 'RTCP 有效时不能更改刀具长度 '

检测时间： 执行程序时。该问题会在第一版 (修订版) 中出现。

引起原因： RTCP 有效时试图换刀。

解决方案： 取消 RTCP，进行换刀，然后再重新激活 RTCP 功能。

1401 ' 先前未取消选择，则不允许使用 #TLC ON'

检测时间： 执行程序时。

引起原因： #TLC ON 已经激活。

解决方案： 核对程序。

1402 '#LINK: 主轴与从轴必须拥有相同的参数 HIRTH 与 HPITCH'

检测时间： 执行程序时。

引起原因： 试图激活一组耦合，其中：

- 主轴为 Hirth 轴，但从轴不是；反之亦然。
- 两根轴均为 Hirth 轴，但参数 HPITCH 不一样。

解决方案： 耦合 (随动) 只能在两根轴同为 Hirth 轴或均不是 Hirth 轴时才有可能。可在机床参数表中更改 Hirth 螺距 (HPITCH)。

1403 '#LINK: Hirth 轴失效时不能定义耦合 (随动)'

检测时间： 执行程序时。

引起原因： 试图在某根 Hirth 轴无效时激活耦合。

解决方案： 激活两根 Hirth 轴，然后再激活耦合。



1404 ' 与 M 功能关联的档不存在 '

检测时间： 执行程序时。

引起原因： M4x (M41 ～ M44) 功能编写了主轴换档，但是 "x" 档不存在。

解决方案： 可用主轴齿轮档数由机床参数NPARSETS给定，因此，CNC只认可M41～M4x("x" 小于或等于 NPARSETS ）功能。

1405 'S 值超出最大档数 '

检测时间： 执行程序时。

引起原因： 为自动换档主轴编写的转速大于主轴最大档时，出现该错误。

解决方案： 可采用的解决办法有：

- 编写较低的主轴转速。
- 增加机床参数表中主轴各档可达到的最大转速。

1406 '#CALL: 不允许编写参数 '

检测时间： 执行程序时。

引起原因： #CALL 指令不允许编写参数。

解决方案： 使用 #PCALL 指令调用子程序。

1407 ' 读型腔数据出错 '

检测时间： 执行程序时。

引起原因： 执行 2D 活 3D 型腔时， CNC 不能解码其中的某些数据。手动编辑型腔数据、用变量替代数字值或删除指令 #DATAP2D、#DATAP3D,... 的括弧时有可能出现该错误。

解决方案： 使用循环编辑器重新编辑型腔。

1408 ' 主轴未控制在指定位置时不允许编程 '

检测时间： 执行程序时。

引起原因： 主轴未控制在指定位置时不能执行编程功能或指令。

解决方案： 主轴需要编码器。

1409 ' 不允许带子程序的 T 功能嵌套 '

检测时间： 执行程序时。

引起原因： 执行 T 功能时，其子程序与机床参数 (TOOLSUB) 关联，该子程序中编写了另一 T 功能。

解决方案： 不能在与 T 功能关联的子程序中编写 T 功能。

1411 '#CD: 程序段号错误 '

检测时间： 执行程序时。

引起原因： 指令中编写的程序段号错误。

解决方案： 可检验的最大程序段号为 200。

1412 '#DGWZ: 图形显示区域定义错误 '

检测时间： 执行程序时。

引起原因： 指令中定义的轴极限错误。

解决方案： 两个极限值均可为正或者负，但是下限必须小于上限。

1413 ' 主轴定位转速不能为 0'

检测时间： 执行程序时。

引起原因： M19 功能的编程定位转速为 0。

解决方案： 编写大于 0 的定位转速，使用语法："S.POS ="。

1414 '#PARK: 该指令只允许使用一根轴 '

检测时间： 执行程序时。

引起原因： #PARK 指令只能停止一根轴。

解决方案： 为各轴分别编写 #PARK 程序段。

1417 ' 文件路径过长 '

检测时间： 执行程序时。
引起原因： 程序或子程序路径的最大字符数超过最大许可值。
程序或子程序路径长度最大为 120 个字符。
解决方案： 将程序或子程序移动至另一个目录下，以缩减路径的字符数。

1418 ' 不能停止主平面或有效转换内的轴 '

检测时间： 执行程序时。
引起原因： 试图停止 (#PARK) 主工作平面或有效运动的轴。
解决方案： 核对程序。

1419 ' 不能停止龙门轴或耦合 (随动) 轴 '

检测时间： 执行程序时。
引起原因： 试图停止 (#PARK) 龙门轴或有效耦合 (#LINK) 轴。
解决方案： 核对程序。

1420 ' 程序末尾有开控制程序段 '

检测时间： 执行程序时。
引起原因： 某些 "$" 控制程序段没有相应的关闭指令。
解决方案： 检查程序与子程序中所有的 "$" 控制程序段：$IF - $ENDIF、 $FOR - $ENFOR、 $SWITCH - $ENDSWITCH……

1421 ' 有效转换的轴不能为从轴，也不能停止 '

检测时间： 执行程序时。
引起原因： 编程转换中停止的轴为龙门对或有效耦合的从轴。
解决方案： 为使用转换中的轴，别让有效转换失效或取消龙门对。

1422 '#CS ON/#ACS ON: 编程模式错误 '

检测时间： 执行程序时。
引起原因： MODE（模式）参数编程错误。
解决方案： MODE（模式）参数的值必须介于 1 ～ 6 之间。

1423 '#CS ON/#ACS ON: 阵列轴的参数必须为 0 或 1'

检测时间： 执行程序时。
引起原因： 阵列轴参数的编程值错误。
解决方案： 阵列轴的参数值必须为 0 或 1。

1424 'MCS 有效时不允许使用 G 功能 '

检测时间： 执行程序时。
引起原因： #MCS 有效时不能执行 G 功能。
解决方案： 核对程序。

1425 ' 跳过程序段只能在每行的开端使用 '

检测时间： 执行程序时。
引起原因： "/" 符号只能在每一行的开端使用。
解决方案： 核对程序。

1426 ' 使用不同的刀具半径加工型腔 '

检测时间： 核对程序。
引起原因： 使用不同的刀具半径加工型腔，且必须重新生成。
解决方案： 重新生成型腔。

1427 ' 轴编程错误 '

检测时间： 执行程序时。
引起原因： 指令或 G 功能编程语法错误，例如：
- G0 <Z>Z20 (轴名重复)。
- G74 X Y (轴的原点搜索顺序缺失)。
- G20 X Y Z (系统中的轴顺序缺失)。

解决方案： 查阅手册，参考指令或 G 功能的语法。

1428 ' 主平面的运动必须编写在循环 G 功能之前 '

检测时间： 执行程序时。

引起原因： 将循环所在平面位置的轴坐标定义在 G 功能之后，因此系统会查询其参数。

解决方案： 将轴坐标编写在循环 G 功能之前。

1429 ' 同一程序段中的子程序过多 '

检测时间： 执行程序时。

引起原因： 同一程序段中的子程序超过最大许可值。最大值为 5。

解决方案： 将子程序编写在不同程序段中，如果有必要，可使用子程序嵌套。

1430 ' 超出数值格式范围 '

检测时间： 执行程序时。

引起原因： 赋予数据、变量或参数的值超出规定格式。

解决方案： 核对程序。

1431 'M19 的主轴位置错误 '

检测时间： 执行程序时。

引起原因： M19 的主轴位置编程值过大。

解决方案： 编写较小的值。

1432 ' 不能编程龙门对或耦合的从轴 '

检测时间： 执行程序时。

引起原因： 在指令 #CALL AX、#SET AX、#FREE AX 或 #RENAME AX 中对有效耦合 (#LINK) 或龙门对的从轴进行了编程。

解决方案： 为对上述指令中的轴进行操作，让有效耦合 (#UNLINK) 失效或取消龙门对 (通过机床参数实现)。

1433 ' 被停止的轴不能为主平面的一部分 '

检测时间： 执行程序时。

引起原因： 使用 #CALL AX 或 #SET AX，将轴配置为系统主平面内的三根轴 之一。

解决方案： 要将轴配置为主平面内三根轴中的一根，首先不能停止它 (#UNPARK)。

1434 ' 配置中不能包含被关联的从轴 '

检测时间： 执行程序时。

引起原因： #CALL AX 或 #SET AX 指令用来在系统配置中包含有效耦合 (#LINK) 或龙门对的主轴。包含主轴的同时也自动包含了从轴，然而从轴决不能占用通道的三个主位置。除了三个主位置，没有其它位置供从轴使用，这就是导致问题的原因。

解决方案： 要包含主轴，必须先让有效耦合失效 (#UNLINK) 或取消龙门对。
要同时包含主轴与从轴，首先从通道中删除另一根轴或增加通道的轴数量。

1435 ' 不能为被关联的从轴赋名，否则将导致重复命名 '

检测时间： 执行程序时。

引起原因： #CALL AX 或 #SET AX 指令用来在系统配置中包含有效耦合 (#LINK) 或龙门对的主轴。包含主轴的同时也自动包含了从轴。导致该问题的原因是从轴名已经被通道内的另一轴占用。

解决方案： 为两根存在冲突的轴中的任意一根重新命名。

1436 ' 程序段搜索中未编写停止程序段 '

检测时间： 执行程序时。

引起原因： 选择了在程序段搜索中执行程序的选项，但是未给出执行操作的最终程序段。

解决方案： 一旦选择程序段搜索选项，将出现停止程序段选项。选择将结束程序段搜索的程序段。

1439 ' 有效转换的轴不能为 Hirth 轴 '

检测时间： 执行程序时。

引起原因： 试图激活轴转换 (#TLC、 #RTCP、 #TOOL ORI、 #CS 或 #ACS)，但是转换中的某根轴为 Hirth 轴。

解决方案： 要成为转换轴，决不能为 HIRTH 轴；因此，设置机床参数 HIRTH = NO。



1441 ' 运动未被激活 '

检测时间： CNC 启动或参数确认时。

引起原因： 某个必须激活的运动没有激活，可能因为它是未知的运动，也有可能因为该运动所需的轴不正确。

解决方案： 确定待激活的运动类型值为有效值，同时确保运动所需的轴正确定义。轴必须为通道主轴，而且它们不能为 Hirth 轴、不能停止且不能为耦合或龙门对的从轴。

1442 ' 运动失效 '

检测时间： 重启期间或程序开头发出警告。

引起原因： 运动失效，可能因为它是未知的运动，也有可能因为该运动所需的轴不正确。

解决方案： 确定待激活的运动类型值为有效值，同时确保运动所需的轴正确定义。轴必须为通道主轴，而且它们不能为 Hirth 轴、不能停止且不能为耦合或龙门对的从轴。

1443 '#CS/ACS 失效 '

检测时间： 重启期间或程序开头发出警告。

引起原因： 斜面坐标转换所需的轴不正确。

解决方案： 确保通道三根主轴已经定义、为直线轴、未停止且不是耦合或龙门对的从轴。

1444 ' 转换的主轴必须为直线轴 '

检测时间： 执行程序时。

引起原因： 编程转换或运动的三根主轴既不是直线轴也不是 C 轴。

解决方案： 转换或运动的三根主轴必须为直线轴 (机床参数 AXISTYPE) 或 C 轴 (机床参数 CAXIS)。

1445 ' 参数值错误 '

检测时间： 执行程序时。

引起原因： 固定循环中的参数赋值错误。

解决方案： 核对程序。

1446 ' 局部子程序中不允许使用起始程序段 '

检测时间： 执行程序时。

引起原因： 起始程序段不能出现在局部子程序中。

解决方案： 在邻近程序段选择另一个起始程序段。

1447 ' 不允许使用软件选项 '

检测时间： 执行程序时。

引起原因： CNC 没有用来执行编程指令的软件选项。

解决方案： 诊断时，可检测 CNC 提供的软件选项。

1448 ' 刀具不能与有效平面正交 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- 主轴类型不允许将刀具置于该位置，因为它有可能是角度轴。
- 定位主轴与有效平面正交可能会超出极限。

解决方案： 定义另一个有效平面或更改主轴。

1449 '#PATH 指令编程错误 '

检测时间： 执行程序时。

引起原因： #PATH 指令语法错误。

解决方案： 参考编程手册。



CNC8070

REF. 0402

ERROR SOLUTIONS

1450 ' 解决方案错误：主轴与有效平面正交 '

检测时间： 执行程序时。

引起原因： 试图读取 TOOLORIF1 或 TOOLORIF2 变量，但是它的值错误，因为刀具不能与有效平面正交。

解决方案： 有好几种原因使得刀具不能与有效平面正交。根据原因的不同，可以采取重新定义新平面或更改主轴的办法解决该问题 (参见 1448 号错误)。

1451 ' 轴类型变量不存在 '

检测时间： 执行程序时。
引起原因： 对所需的轴类型（直线、旋转或主轴），该变量不存在。
解决方案： 核对程序。

1452 ' 驱动器类型变量不存在 '

检测时间： 执行程序时。
引起原因： 对所需的驱动器类型 (模拟、仿真或 SERCOS)，该变量不存在。
解决方案： 核对程序。

1453 ' 轴名过长 '

检测时间： 执行程序时。
引起原因： 轴名多于两个字符。
解决方案： 有效轴名如下所示：
X、 X1 ... X9、 Y、 Y1、 ... Y9、 ...、 Z、 Z1、 ... Z9
A、 A1 ... A9、 B、 B1、 ... B9、 ...、 C、 C1、 ... C9
U、 U1 ... U9、 V、 V1、 ... V9、 ...、 W、 W1、 ... W9
S、 S1 ... S9

1455 'PROFILE: 轮廓无效 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
- 在循环编辑器的轮廓加工循环中，包含轮廓加工循环的文件丢失。
- 循环编辑器轮廓加工循环中指定的文件为空。

解决方案： 循环编辑器的轮廓加工循环必须指定包含轮廓的文件。

1456 '#POLY: 参数缺失 '

检测时间： 执行程序时。
引起原因： #POLY 指令中缺失必不可少的参数 "EP" 或 "SP"。
解决方案： 查阅编程手册，检查指令语法。

1457 '#POLY: 参数值错误 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
- 多项式插补参数错误。
- 曲率半径小于或等于 0。

解决方案： 多项式的插补参数必须为正， SP < EP 且曲率半径必须大于 0。

1458 '#POLY: 编程轴过多 '

检测时间： 执行程序时。
引起原因： 多项式中编写的轴超过 3 根。
解决方案： 多项式插补只能编写 3 根轴。

1459 '#POLY: 起点错误 '

检测时间： 执行程序时。
引起原因： 多项式的起点与前一位置不一样。
解决方案： 更改多项式每根轴的独立项，使之与前一程序段的终点位置一致。

1461 'G9: 弧中点编程错误 '

检测时间： 执行程序时。
引起原因： 弧中点的一个或两个坐标未编程。
解决方案： G9 功能需要编写弧中点的两个坐标。

1462 'G8: 不能计算切线轨迹 '

检测时间： 执行程序时。
引起原因： 不能根据编程半径与终点计算出前一轨迹的圆弧切线。
解决方案： 检查编程半径与终点的值。



1463 'G9: 圆轨迹编程错误 '

检测时间： 执行程序时。
引起原因： 通过该三点不能形成一个圆。
解决方案： 在程序段中定义两点，与前一运动的终点共同定义一个圆弧。记住三点不能相同且不能在同一直线上。

1464 ' 旋转轴超出模块范围 '

检测时间： 执行程序时。
引起原因： 为 MODULE（模块）类旋转轴编写的绝对坐标 (G90) 错误。
解决方案： 轴坐标必须介于机床参数 MODUPLIM 与 MODLOWLIM 设定的极限之间。

1465 'RTCP 与 TLC 功能彼此不相容 '

检测时间： 执行程序时。
引起原因： 试图在其中一个功能有效时激活另一个功能。
解决方案： 不能同时激活两个功能。

1466 ' 不能将 G8 作为 G36/G37/G38/G39 的第二运动程序段 '

检测时间： 执行程序时。
引起原因： G8 程序段不能作为 G36/G37/G38/G39 功能的第二运动程序段。一方面，这些功能并没有可连接两个运动程序段的中间程序段，另一方面，前面没有程序段可供 G8 相切。
解决方案： G36/G37/G38/G39 功能的第二程序段必须为 G0/G1/G2/G3。

1467 'POSLIMIT/NEGLIMIT 不能超出机床参数值 '

检测时间： 执行程序时。
引起原因： 写入变量 A.POSLIMIT.axis 和 A.NEGLIMIT.axis 中的值不能超过轴机床参数 POSLIMIT 和 NEGLIMIT 的值。
解决方案： 核对程序。

1468 'G30: 极点编程错误 '

检测时间： 执行程序时。
引起原因： 只编写了极点的一个坐标。
解决方案： 必须在两根轴上编写极点坐标。

1469 ' 半径不能为负或 0'

检测时间： 执行程序时。
引起原因： 以极坐标编写了运动，但是极半径为负或 0。
解决方案： 极坐标必须大于0。在增量模式中，编程值可为负或0，但是绝对极半径不可以这样。

1470 'UNIDIR 旋转轴增量模式编程错误 '

检测时间： 执行程序时。
引起原因： 为 MODULE 和 UNIDIR 类旋转轴编写的增量坐标错误。
解决方案： 如果旋转轴的机床参数 UNIDIR 为正 (POSITIVE)，则增量坐标必须为正值或 0 。如果旋转轴的机床参数 UNIDIR 为负 (NEGATIVE)， 则增量坐标必须为负值或 0。

1471 'G73: 旋转中心编程错误 '

检测时间： 执行程序时。
引起原因： 只编写了图案（坐标系统）旋转的一个坐标。
解决方案： 必须在两根轴上编写旋转中心坐标。

1472 'G73: 未编写旋转角度 '

检测时间： 执行程序时。
引起原因： 编写了旋转中心坐标，但是未编写旋转角度。
解决方案： 必须在两根轴上同时编写旋转中心坐标与旋转角度。

1473 ' 图案旋转有效时不能编写 #POLY 指令 '

检测时间： 执行程序时。
引起原因： 在坐标系统旋转 (G73) 有效时编写了 #POLY 指令。
解决方案： 核对程序。

1475 ' 半径重复编程 '

检测时间： 执行程序时。

引起原因： 半径 "R" 或 "R1" 在同一程序段中编写了一次以上。

解决方案： 在一个程序段中只能定义一个半径。

1476 ' 使用不同的刀尖半径生成型腔 '

检测时间： 执行程序时。

引起原因： 生成型腔的刀具半径不是当前的刀具半径。要执行该操作，必须重新生成。

解决方案： 重新生成型腔。

1477 ' 使用不同的刀具切削长度生成型腔 '

检测时间： 执行程序时。

引起原因： 生成型腔的刀具切削长度并非当前的刀具切削长度。要执行该操作，必须重新生成。

解决方案： 重新生成型腔。

1478 ' 使用不同的刀具切入角生成型腔 '

检测时间： 执行程序时。

引起原因： 生成型腔的刀具切入角并非当前的刀具切入角。要执行该操作，必须重新生成。

解决方案： 重新生成型腔。

1479 'G74: 未关联子程序 '

检测时间： 执行程序时。

引起原因： G74 功能位于单独程序段中，但是未定义由机床参数关联的子程序。

解决方案： 使用机床参数 REFSUB 为 G74 指定子程序。

1480 ' 编程 : #EXEC [" 轨迹 + 程序 ", 通道]'

检测时间： 执行程序时。

引起原因： 可能的原因包括：
- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1481 ' 通道号错误 '

检测时间： 执行程序时。

引起原因： 指令 #EXEC、 #MEET 或 #WAIT 中编写的通道号错误。

解决方案： 通道号必须介于 1 ～ 4 之间。

1483 ' 编程 : #WAIT/#MEET [信号、通道、通道、. . .]'

检测时间： 执行程序时。

引起原因： 可能的原因包括：
- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1484 ' 信号数超出规定范围 '

检测时间： 执行程序时。

引起原因： 指令 #WAIT、 #MEET 或 #SIGNAL 中编写的信号数错误。

解决方案： 指令 #WAIT、 #MEET 和 #SIGNAL 中编写的信号数必须介于 1 ～ 10 之间。

1485 '#WAIT/#MEET 未生效 '

检测时间： 执行程序时。

引起原因： 指令 #WAIT 或 #MEET 未形成任何等待，因为同一通道中编写了信号延时激活 。

解决方案： 核对程序。

1486 ' 编程 : #SIGNAL [信号、信号、信号、. . .]'

检测时间： 执行程序时。

引起原因： 可能的原因包括：
- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1487 ' 编程 : #CLEAR [信号、信号、信号、. . .]'

检测时间： 执行程序时。

引起原因： 可能的原因包括：
- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1489 ' 结果组中轴名重复 '

检测时间： 执行程序时。

引起原因： 使用指令 #RENAME AX 将多根轴重新命名为相同的名字。

解决方案： 以不同的名字为每根轴进行重新命名，以免同一通道中两根轴具有相同的名字。

1490 'G63 前需要编写 M19'

检测时间： 执行程序时。

引起原因： 试图使用 SERCOS 主轴进行 G63 加工螺纹，但是前面没有使用 M19 进行定位。

解决方案： 在进行攻丝前编写 M19。

1491 ' 探头号错误 '

检测时间： 执行程序时。

引起原因： #SELECT PROBE 功能选择的探头号错误。

解决方案： 选择的探头号必须为 1 或 2。

1492 ' 没有与探头 (PRBDI1/2) 关联的数字输入 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：
- G100 功能正在进行测量，但是没有与探头关联的数字探头输入。
- #SELECT PROBE 指令选择了探头，但是没有与探头关联的数字输入。

解决方案： 使用机床参数 PRBDI1/2 为探头关联数字输入。

1493 ' 不能同时编写 #SPLINE ON、 G41/G42 和 G136'

检测时间： 执行程序时。

引起原因： 试图在 CNC 中同时激活 #SPLINE ON、 G41、 G42 和 G136 功能。

解决方案： 取消里面的某些功能。

1494 ' 索引错误 '

检测时间： 执行程序时。

引起原因： #ERROR 或 #WARNING 指令中给出的错号不存在。

解决方案： 核对程序。

1495 '#PROBE1: 探测循环的轴错误 '

检测时间： 执行程序时。

引起原因： PROBE1 循环诱发了该错误，因为当前配置的三根主轴与初始配置中的不同。

解决方案： 将通道中的第一个三根轴返回初始配置。

1496 '#PROBE1: #TOOL AX[-] 有效时不允许编程 '

检测时间： 执行程序时。

引起原因： 试图执行 PROBE1 循环，此时刀具定位在轴的负方向上。

解决方案： 在执行 PROBE1 循环前编写 #TOOL AX[+]，将刀具定位在轴的正方向上。



1497 ' 变量类型算子错误 '

检测时间： 执行程序时。

引起原因： 试图使用算子 "+="、"-="、"*=" y "/=" 改变变量 AXISNAMEi、GAXISNAMEi、SPDLNAMEi 或 GSPDLNAMEi 的值。

解决方案： 核对程序。

1499 '#RPT 与子程序的嵌套层过多 '

检测时间： 执行程序时。

引起原因： 两个标签之间的 #RPT 指令定义了另一条 #RPT 指令的影响区域。导致该错误的原因是 RPT 指令与子程序的嵌套层数超出了规定范围。

解决方案： 嵌套的最大许可层数为 20。

1500 '#EXEC: 不能在指定通道执行程序 '

检测时间： 执行程序时。

引起原因： 使用 #EXEC 在通道中执行程序，该通道：

- 正在执行另一个程序。
- 正处于出错状态。
- 正处于手动状态且不能切换至自动模式。

解决方案： 等待在另一个通道中执行该程序，或进行重启。

1501 '#RPT 中的标签重复 '

检测时间： 执行程序时。

引起原因： #RPT 指令中的起始与结束标签相同。

解决方案： 定义不同的起始与结束标签。

1502 ' 变量需要编写数组索引 '

检测时间： 执行程序时。

引起原因： 编写了变量，但是未给出读和写变量的索引。

解决方案： 参考编程手册中的控制变量。

1503 ' 变量需要对轴进行编程 '

检测时间： 执行程序时。

引起原因： 编写了轴变量，但是未给出轴索引。

解决方案： 参考编程手册中的控制变量。

1504 ' 变量不允许对数组索引进行编程 '

检测时间： 执行程序时。

引起原因： 读或写非数组变量时，编写了数组索引。

解决方案： 参考编程手册中的控制变量。

1505 ' 变量不允许对轴进行编程 '

检测时间： 执行程序时。

引起原因： 读或写非轴变量时，编写了轴。

解决方案： 参考编程手册中的控制变量。

1506 ' 标签未定义或 #RPT 指令标签交换 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- 第二个标签未定义。
- 第一个标签与第二个标签之间编写了 M30 指令。
- #RPT 指令中的第二个标签编程为第一个，从而成为第一个标签。

解决方案： 正确定义标签与指令。

1507 '#RPT 的第二个标签必须编写的单独程序段中 '

检测时间： 执行程序时。

引起原因： 编写 #RPT[N1,N2] 指令时，第二个标签必须编写在单独程序段中，该程序段中不能有任何其他类型的指令 。如果编写 N2: X10，则会出现该问题。

解决方案： 结束标签必须编写在单独程序段中。在前一程序段编写该指令并与 #RPT 执行，或者 在下一行编写，不与 #RPT 一块执行。



1508 ' 主平面轴不允许使用 G201 和有效 C 轴 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

1. C 轴编写了 G201 功能，该轴同时还是通道三根主轴中的一根。
2. #FACE 指令中编写的轴为附加手动模式， G201。

解决方案： 可采用的解决办法有：

1. 编写 #FACE OFF，使该轴不再为 C 轴。
2. 编写 G202 #AXIS [axis]，使该轴不再处于附加手动模式中。

1509 '#SET AX/#CALL AX: 偏置编程无效 '

检测时间： 执行程序时。

引起原因： 指令中的所有轴均已属于当前配置，因此，该指令仅仅意味着次序的改变。在此情形下，编程的偏置选项无效。

解决方案： 核对程序。

1510 ' 轴在通道中不存在或不可用 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- 试图移动通道中在不存在或不可用的轴。
- 所执行指令中的轴在通道中不存在或不可用。
- 试图对通道中不存在或不可用的轴变量进行读或写操作。

解决方案： 预先指定属于该通道的轴。

可通过更改通道的机床参数 CHNAXIS 和 CHAXISNAME 或者使用 #SET AX 或 #CALL AX 指令使通道包含一根新轴。

1511 ' 刀具不在刀库中 '

检测时间： 执行程序时。

引起原因： 通过 TM.P[ ] 变量读取刀具在刀库中的位置，但是刀具不在刀库中。

解决方案： 只能读取位于刀库中的刀具的位置。

1512 ' 释放刀库位置或刀库位置不存在 '

检测时间： 执行程序时。

引起原因： 通过 TM.T[ ] 变量读取刀号，但是刀具的刀库位置不存在。

解决方案： 只能读取位于刀库有效位置的刀具。

1513 ' 写变量出错 '

检测时间： 执行程序时。

引起原因： 试图写变量，而：

- 该变量不存在。
- 该变量为轴变量，但是轴不存在。
- 该变量为轴变量，但是该轴类型不存在。
- 该变量为轴变量，但是该轴驱动类型不存在。
- 该变量不允许使用复合算子 (+=...) 。
- 写入值对该变量无效。

解决方案： 检查变量语法以及写入值。

1514 'G72 有效时不能生成 2D-3D 型腔 '

检测时间： 执行程序时。

引起原因： 试图生成型腔时，比例缩放因子 (G72 或 #SCALE[n] 指令) 已经被激活。

解决方案： 取消比例缩放因子。

1515 ' 初始配置的一根或几根轴不可用 '

检测时间： 执行程序时。

引起原因： 临时交换轴被释放，且不能通过重启或在程序开头恢复，因为通道没有释放该轴。

解决方案： 其他通道可通过重启或在另一程序的开头释放该轴。同时也可通过 #FREE AX 指令释放该轴。



1516 ' 需要输入值 '

检测时间： 执行程序时。

引起原因： 子程序调用参数列表 (与 #PCALL 或 G180-189 一起) 编程错误。

解决方案： 调用子程序的参数列表由以下部分组成：

- 局部算术参数：P0 = 30。
- 字母：A = 30 或 A30。

两种可在列表中混合使用。

1518 'NR 需要在程序段中编写运动 '

检测时间： 执行程序时。

引起原因： 程序段重复 (NR) 未包含运动。

解决方案： 使用 NR 编写程序段重复，只有与包含运动的程序段在一起才有效。其他类型的程序段重复必须使用由 CNC 提供的流程控制指令 。

1519 'NR: 同一程序段中不能编写 M/T/D/H'

检测时间： 执行程序时。

引起原因： 重复次数 (NR) 所在的程序段编写了 M、 T、 D 或 H 功能。

解决方案： 使用 NR 编写程序段重复，只有与包含运动的程序段在一起才有效。其他类型的程序段重复必须使用由 CNC 提供的流程控制指令 。

1520 'NR: 同一程序段中不能编写 $GOTO 指令 '

检测时间： 执行程序时。

引起原因： 重复次数 (NR) 所在的程序段编写了 $GOTO 指令。

解决方案： 使用 NR 编写程序段重复，只有与包含运动的程序段在一起才有效。其他类型的程序段重复必须使用由 CNC 提供的流程控制指令 。

1521 'NR: 同一程序段中不能调用子程序 '

检测时间： 执行程序时。

引起原因： 重复次数 (NR) 所在的程序段编写了子程序调用 (L、 LL、 #CALL、 #PCALL、 #MCALL 或 G180-G189)。

解决方案： 使用 NR 编写程序段重复，只有与包含运动的程序段在一起才有效。其他类型的程序段重复必须使用由 CNC 提供的流程控制指令 。

1522 ' 不允许使用负值 '

检测时间： 执行程序时。

引起原因： 编写的程序段重复次数 (NR) 为负。

解决方案： 编写正值或 0。

1523 'POS 与 T 必须位于同一程序段中 '

检测时间： 执行程序时。

引起原因： 刀具 T 与位置 POS 未在同一程序段中。

解决方案： 刀具 T 及其在刀库中的位置必须位于同一程序段中。

1525 ' 轴不能交换 '

检测时间： 执行程序时。

引起原因： 当机床参数 "AXISEXCH" = NO 时，试图在指令 #CALL AX、 #SET AX 或 #FREE AX 中交换轴。

解决方案： 要在通道间交换轴，这些轴的机床参数 "AXISEXCH" 必须设为 "temporary (临时)" 或 "maintained (继续保持)"。

1526 '#EXEC: 指定的通道并非 CNC 通道 '

检测时间： 执行程序时。

引起原因： 使用 #EXEC 指令在通道中执行程序，但是该通道不是 CNC 通道，而是 PLC 通道。

解决方案： 可使用机床参数 CHTYPE 改变通道类型。

1527 '% 后的标识符错误 '

检测时间： 执行程序时。
引起原因： 在 #MSG、 #ERROR 或 #WARNING 指令中， "%" 后编写的标识符错误。
解决方案： 有效标识符为：

- 显示数字：%D 或 %d。
- 显示 "%" 字符：%%。

1529 ' 右引号后需要标识符列表或 "]"'

检测时间： 执行程序时。
引起原因： 在 #MSG、 #ERROR 或 #WARNING 指令中，需显示的信息后编写的字符无效。
解决方案： 在需要显示的信息的右引号后，只能辨识由逗号分隔的变量列表以及右方括弧。查阅编程手册，核对该指令的语法。

1530 'G53 不能采用极坐标编程 '

检测时间： 执行程序时。
引起原因： G53 功能中的终点由极坐标或圆柱坐标定义。
解决方案： 以机床原点为参考进行编程时，必须使用笛卡儿坐标来定义点。

1531 ' 程序 : #EXBLK [程序段、通道]'

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1532 ' 程序 : #MASTER < 主轴名 >'

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1533 ' 程序 : #FREESP [sp1、 sp2、 ...]'

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1534 ' 程序 : #CALLSP [sp1、 sp2、 ...]'

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1535 ' 程序 : #SETSP [sp1、 sp2、. .]'

检测时间： 执行程序时。
引起原因： 可能的原因包括：

- 指令语法错误。
- 只有标签的程序段号才能与该指令位于同一程序段中。

解决方案： 对于第一种情形，可参考编程手册。对于第二种情形，将它们编写在不同程序段中 。

1538 ' 主轴不能从通道中恢复 '

检测时间： 执行程序时。
引起原因： 如果属于某个通道的主轴被释放，且正被另一个通道使用，那么不能通过重启或在程序开头恢复 (AXISEXCH = TEMPORARY（临时))。
解决方案： 使用该主轴的通道必须释放它。否则，该主轴原来所在的通道不能使用它，而且在每次重启或在程序的开头都会发出警告。



1539 '结果组中主轴名重复'

检测时间： 执行程序时。

引起原因： 使用指令 #RENAME SP 将多根主轴重新命名为相同的名字。

解决方案： 以不同的名字为每根主轴重新命名，以免同一通道中两根轴具有相同的名字。

1540 '通道中无主轴时不允许编程'

检测时间： 执行程序时。

引起原因： 可能的原因包括：

- 试图读或写主轴变量，但是主轴在通道中不存在。
- 如果通道中无主轴，则不能执行 G 功能或指令。

解决方案： 使用 #MASTER 指令为通道定义主轴。

1541 'C 轴有效时不能删除主轴'

检测时间： 执行程序时。

引起原因： 试图使用 #FREE SP 或 #SET SP 指令删除作为 C 轴工作的主轴。

解决方案： 从该通道中移除主轴前，使用 #CAX OFF 指令取消 C 轴，或激活另一主轴为 C 轴。

1542 '不能交换主轴'

检测时间： 执行程序时。

引起原因： 使用 #CALL SP、#SET SP 或 #FREE SP 指令将主轴从一个通道换至另一个通道，但是该主轴不可交换。

解决方案： 要使主轴可交换，设置其机床参数 AXISEXCH 为 TEMPORARY（暂时）或 MAINTAINED（继续保持）。

1544 '不能在同一程序段中使用 G63 和 M3/M4/M5/M19/M41-M44'

检测时间： 执行程序时。

引起原因： 刚性攻丝 G63 所在程序段不能使用与主轴相关的 M 功能。

解决方案： 根据所需获得的结果，可以将其编写在前一程序段或后一程序段。如果编写在下一程序段，那么这些 M 功能将取消模态指令 G63，如果需要继续攻丝，则必须重新编写 G63。

1545 'G63 或 #CAX 有效时不能进行主轴换档'

检测时间： 执行程序时。

引起原因： G63 攻丝有效或主轴作为 C 轴使用时不能进行换档。

解决方案： 取消 G63 或 #CAX 后换档，如果有必要，则重新激活 G63 功能或 #CAX。

1546 '主轴未挂档时不允许使用 G63'

检测时间： 执行程序时。

引起原因： 在系统启动或重启后，如果 PLC 未使用 GEAR1/2/3/4 指定主轴档数，则会出现该问题。此时主轴未挂档，如果在编写 G63 前未使用主轴，也不能自动生成档数。

解决方案： 启动 G63 攻丝前，主轴必须挂档。激活 G63 前编写主轴换档。

- 直接使用 M41-M44 功能。
- 如果主轴为 "AUTOGEAR（自动挂档）"，编写主轴转速 (S) 时自动挂档。
- 如果主轴不是"AUTOGEAR（自动挂档）"，可通过编写主轴转速(S)以及M3或M4进行挂档。

1547 'G63 有效时不允许使用 #CAX'

检测时间： 执行程序时。

引起原因： G63 功能使用的主轴不能激活为 C 轴。

解决方案： 编写 #CAX 前取消 G63 或者使用通道内的另一根主轴为 C 轴。

1548 'G33/G63/G95/G96 有效时不能改变主轴'

检测时间： 执行程序时。

引起原因： G63 和 G96 功能使用通道主轴。在上述功能有效时不能从通道中移除主轴。

解决方案： 释放主轴前先取消 G63 或 G96。



1549 ' 型腔名为空或错误 '

检测时间： 执行程序时。

引起原因： 使用循环编辑器定义 2D 或 3D 型腔时，未定义型腔名或型腔名错误。循环编辑器生成的型腔名不能为 #DATAP2D 或 #DATAP3D。

解决方案： 为型腔指定其他的名字。

1550 ' 通道中不能编写两根 C 轴 '

检测时间： 执行程序时。

引起原因： 可使用通道内的任何轴激活 C 轴功能，但是不能同时激活两根。

解决方案： 在将其激活为另一根 C 轴前取消主轴上的 #CAX。

1551 ' 局部参数嵌套层过多 '

检测时间： 执行程序时。

引起原因： #PCALL 或 G1-G190 功能的子程序调用嵌套增加了局部参数的嵌套层。原因是超出了局部变量的最大许可嵌套层 (7)。

解决方案： 减少子程序嵌套或使用 #CALL、 L 或 LL 指令调用它们；它们不会增加局部参数的嵌套层数。

1552 ' 变量名必须以 "V." 开头 '

检测时间： 执行程序时。

引起原因： 工件程序或 MDI 中的变量名没有前缀 "V."。

解决方案： 为变量添加前缀 "V."。

1553 ' 用户变量过多 '

检测时间： 执行程序时。

引起原因： 不能超出可定义的最大用户变量 (V.P、 V.S) 数。

解决方案： 可定义的用户变量最多为 20 个。

1554 'PLC 不能辨识 #EXEC 指令中的 START（启动）'

检测时间： 执行 #EXEC 指令时。

引起原因： 发出指令在另一通道中执行程序时， PLC 必须将 "go ahead（跳至）" 发送至程序启动部分，如果未发送该指令，则出现该错误。

解决方案： 检查通道中所有可能通过 PLC 妨碍程序执行的因素。

1700 ' 中心钻孔： F = 0'

检测时间： 执行中心钻孔循环时。

引起原因： 循环中编写的 F 为 0。

解决方案： 为其赋予大于 0 的值。

1701 ' 中心钻孔： S = 0'

检测时间： 执行中心钻孔循环时。

引起原因： 循环中编写的 S 为 0。

解决方案： 为其赋予大于 0 的值。

1702 ' 中心钻孔： T = 0'

检测时间： 执行中心钻孔循环时。

引起原因： 循环中未编写刀具 T。

解决方案： 指定刀具 T。

1703 ' 中心钻孔： P = 0'

检测时间： 执行中心钻孔循环时。

引起原因： 循环中编写的 P (深度) 等于 0。

解决方案： 为其赋予大于 0 的值。

1704 ' 中心钻孔： ALPHA = 0'

检测时间： 执行中心钻孔循环时。

引起原因： 循环中编写的角度等于 0。

解决方案： 为其赋予大于 0 的值。



1705　' 中心钻孔： 直径 = 0'

检测时间：　执行中心钻孔循环时。
引起原因：　循环中编写的直径等于 0。
解决方案：　为其赋予大于 0 的值。

1706　' 钻孔 1: F = 0'

检测时间：　执行钻孔循环 1 时。
引起原因：　循环中编写的 F 等于 0。
解决方案：　为其赋予大于 0 的值。

1707　' 钻孔 1: S = 0'

检测时间：　执行钻孔循环 1 时。
引起原因：　S 循环中编写的 S 等于 0。
解决方案：　为其赋予大于 0 的值。

1708　' 钻孔 1: T = 0'

检测时间：　执行钻孔循环 1 时。
引起原因：　循环中未编写刀具 T。
解决方案：　指定刀具 T。

1709　' 钻孔 1: P = 0'

检测时间：　执行钻孔循环 1 时。
引起原因：　循环中编写的 P (深度) 等于 0。
解决方案：　为其赋予大于 0 的值。

1710　' 钻孔 2: F = 0'

检测时间：　执行钻孔循环 2 时。
引起原因：　循环中编写的 F 等于 0。
解决方案：　为其赋予大于 0 的值。

1711　' 钻孔 2: S = 0'

检测时间：　执行钻孔循环 2 时。
引起原因：　循环中编写的 S 等于 0。
解决方案：　为其赋予大于 0 的值。

1712　' 钻孔 2: T = 0'

检测时间：　执行钻孔循环 2 时。
引起原因：　循环中未编写刀具 T。
解决方案：　指定刀具 T。

1713　' 钻孔 2: P = 0'

检测时间：　执行钻孔循环 2 时。
引起原因：　循环中编写的 P (深度) 等于 0。
解决方案：　为其赋予大于 0 的值。

1714　' 钻孔 2: B = 0'

检测时间：　执行钻孔循环 2 时。
引起原因：　循环中编写的 B (单次钻孔深度) 等于 0。
解决方案：　为其赋予大于 0 的值。

1715　' 加工螺纹循环： F = 0'

检测时间：　执行加工螺纹循环时。
引起原因：　循环中编写的 F 等于 0。
解决方案：　为其赋予大于 0 的值。



1716 ' 加工螺纹循环： S = 0'

检测时间: 执行加工螺纹循环时。
引起原因: 循环中编写的 S 等于 0。
解决方案: 为其赋予大于 0 的值。

1717 ' 加工螺纹循环： T = 0'

检测时间: 执行加工螺纹循环时。
引起原因: 循环中未编写刀具 T。
解决方案: 指定刀具 T。

1718 ' 加工螺纹循环： P = 0'

检测时间: 执行加工螺纹循环时。
引起原因: 循环中编写的 P (深度) 等于 0。
解决方案: 为其赋予大于 0 的值。

1719 ' 铰孔： F = 0'

检测时间: 执行铰孔循环时。
引起原因: 循环中编写的 F 等于 0。
解决方案: 为其赋予大于 0 的值。

1720 ' 铰孔： S = 0'

检测时间: 执行铰孔循环时。
引起原因: 循环中编写的 S 等于 0。
解决方案: 为其赋予大于 0 的值。

1721 ' 铰孔： T = 0'

检测时间: 执行铰孔循环时。
引起原因: 循环中未编写刀具 T。
解决方案: 指定刀具 T。

1722 ' 铰孔： P = 0'

检测时间: 执行铰孔循环时。
引起原因: 循环中编写的 P (深度) 等于 0。
解决方案: 为其赋予大于 0 的值。

1723 ' 镗孔 1: F = 0'

检测时间: 执行镗孔循环 1 时。
引起原因: 循环中编写的 F 等于 0。
解决方案: 为其赋予大于 0 的值。

1724 ' 镗孔 1: S = 0'

检测时间: 执行镗孔循环 1 时。
引起原因: 循环中编写的 S 等于 0。
解决方案: 为其赋予大于 0 的值。

1725 ' 镗孔 1: T = 0'

检测时间: 执行镗孔循环 1 时。
引起原因: 循环中未编写刀具 T。
解决方案: 指定刀具 T。

1726 ' 镗孔 1: P = 0'

检测时间: 执行镗孔循环 1 时。
引起原因: 循环中编写的 P (深度) 等于 0。
解决方案: 为其赋予大于 0 的值。



1727 ' 钻孔 3: F = 0'

检测时间： 执行钻孔循环 3 时。
引起原因： 循环中编写的 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1728 ' 钻孔 3: S = 0'

检测时间： 执行钻孔循环 3 时。
引起原因： 循环中编写的 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1729 ' 钻孔 3: T = 0'

检测时间： 执行钻孔循环 3 时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定刀具 T。

1730 ' 钻孔 3: P = 0'

检测时间： 执行钻孔循环 3 时。
引起原因： 循环中编写的 P (深度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1731 ' 镗孔 2: F = 0'

检测时间： 执行镗孔循环 1 时。
引起原因： 循环中编写的 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1732 ' 镗孔 2: S = 0'

检测时间： 执行镗孔循环 1 时。
引起原因： 循环中编写的 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1733 ' 镗孔 2: T = 0'

检测时间： 执行镗孔循环 1 时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定刀具 T。

1734 ' 镗孔 2: P = 0'

检测时间： 执行镗孔循环 1 时。
引起原因： 循环中编写的 P (深度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1735 ' 矩形型腔 : F = 0'

检测时间： 执行矩形型腔循环时。
引起原因： 循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1736 ' 矩形型腔 : S = 0'

检测时间： 执行矩形型腔循环时。
引起原因： 循环中编写的粗加工或精加工 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1737 ' 矩形型腔 : T = 0'

检测时间： 执行矩形型腔循环时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定粗加工或精加工刀具 T。

1738 **' 矩形型腔： P = 0'**

检测时间： 执行矩形型腔循环时。
引起原因： 循环中编写的 P (型腔深度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1739 **' 矩形型腔： 刀具直径小于 δ '**

检测时间： 执行矩形型腔循环时。
引起原因： 粗加工或精加工刀具直径小于主平面内的粗加工或精加工轨迹。
解决方案： 缩小 XY (δ) 方向的铣削加工轨迹，或者在机床上安装较大直径的刀具。

1740 **' 矩形型腔： 刀具直径大于型腔 '**

检测时间： 执行矩形型腔循环时。
引起原因： 粗加工或精加工刀具的直径大于型腔尺寸 L 或 H。
解决方案： 选择直径较小的刀具加工型腔。

1741 **' 矩形型腔： 精加工刀具直径小于 δ '**

检测时间： 执行矩形型腔循环时。
引起原因： 精加工刀具的直径小于主平面内的精加工余量 (δ)。
解决方案： 选择直径较大的刀具加工型腔。

1742 **' 预空型腔： F = 0'**

检测时间： 执行预空型腔循环时。
引起原因： 循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1743 **' 预空型腔： S = 0'**

检测时间： 执行预空型腔循环时。
引起原因： 循环中编写的粗加工或精加工 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1744 **' 预空型腔： T = 0'**

检测时间： 执行预空型腔循环时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定粗加工或精加工刀具 T。

1745 **' 预空型腔： P = 0'**

检测时间： 执行预空型腔循环时。
引起原因： 循环中编写的 P (型腔深度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1746 **' 预空型腔： 刀具直径小于 δ '**

检测时间： 执行预空型腔循环时。
引起原因： 粗加工或精加工刀具直径小于主平面内的粗加工或精加工轨迹。
解决方案： 缩小 XY (δ) 方向的铣削加工轨迹，或者在机床上安装较大直径的刀具。

1747 **' 预空型腔： 刀具直径大于型腔 '**

检测时间： 执行预空型腔循环时。
引起原因： 粗加工或精加工刀具的直径大于型腔尺寸 L 或 H。
解决方案： 选择直径较小的刀具加工型腔。

1748 **' 预空型腔： 精加工刀具直径小于 δ '**

检测时间： 执行预空型腔循环时。
引起原因： 精加工刀具的直径小于主平面内的精加工余量 (δ)。
解决方案： 选择直径较大的刀具加工型腔。

1749 '预空型腔：R <r'

检测时间： 执行预空型腔循环时。
引起原因： 型腔外直径 (R) 小于内半径 (r)。
解决方案： 设置参数 R，使 R >r。

1750 '矩形凸台：F = 0'

检测时间： 执行矩形凸台循环时。
引起原因： 循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1751 '矩形凸台：S = 0'

检测时间： 执行矩形凸台循环时。
引起原因： 循环中编写的粗加工或精加工 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1752 '矩形凸台：T = 0'

检测时间： 执行矩形凸台循环时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定粗加工或精加工刀具 T。

1753 '矩形凸台：P = 0'

检测时间： 执行矩形凸台循环时。
引起原因： 循环中编写的 P (凸台高度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1754 '矩形凸台：刀具直径小于 δ '

检测时间： 执行矩形凸台循环时。
引起原因： 粗加工或精加工刀具直径小于主平面内的粗加工或精加工轨迹。
解决方案： 缩小 XY (δ) 方向的铣削加工轨迹，或者在机床上安装较大直径的刀具。

1755 '矩形凸台：精加工刀具直径小于 δ '

检测时间： 执行矩形凸台循环时。
引起原因： 精加工刀具的直径小于主平面内的精加工余量 (δ)。
解决方案： 选择直径较大的刀具加工矩形凸台。

1756 '圆形凸台：F = 0'

检测时间： 执行圆形凸台循环时。
引起原因： 循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1757 '圆形凸台：S = 0'

检测时间： 执行圆形凸台循环时。
引起原因： 循环中编写的粗加工或精加工 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1758 '圆形凸台：T = 0'

检测时间： 执行圆形凸台循环时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定粗加工或精加工刀具 T。

1759 '圆形凸台：P = 0'

检测时间： 执行圆形凸台循环时。
引起原因： 循环中编写的 P (凸台高度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1760 '圆形凸台：R = 0'

检测时间： 执行圆形凸台循环时。
引起原因： 循环中编写的 R (凸台半径) 等于 0。
解决方案： 为其赋予大于 0 的值。

1761 '圆形凸台：刀具直径小于 δ '

检测时间： 执行圆形凸台循环时。
引起原因： 粗加工或精加工刀具直径小于主平面内的粗加工或精加工轨迹。
解决方案： 缩小 XY (δ) 方向的铣削加工轨迹，或者在机床上安装较大直径的刀具。

1762 '圆形凸台：精加工刀具直径小于 δ '

检测时间： 执行圆形凸台循环时。
引起原因： 精加工刀具的直径小于 XY 平面内的精加工余量 (δ)。
解决方案： 选择直径较大的刀具加工圆形凸台。

1763 '平面铣削：F = 0'

检测时间： 执行平面铣削循环时。
引起原因： 循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1764 '平面铣削：S = 0'

检测时间： 执行平面铣削循环时。
引起原因： 循环中编写的粗加工或精加工 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1765 '平面铣削：T = 0'

检测时间： 执行平面铣削循环时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定粗加工或精加工刀具 T。

1766 '平面铣削：P = 0'

检测时间： 执行平面铣削循环时。
引起原因： 循环编写的 P (平面铣削深度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1767 '平面铣削：L = 0 与 H = 0'

检测时间： 执行平面铣削循环时。
引起原因： 循环中编写的两个尺寸 (长度与宽度) 等于 0。
解决方案： 至少为两个尺寸里的一个（L 或 H）赋予大于 0 的值。

1768 '点到点仿形：F = 0'

检测时间： 执行点到点仿形循环时。
引起原因： 循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1769 '点到点仿形：S = 0'

检测时间： 执行点到点仿形循环时。
引起原因： 循环中编写的粗加工或精加工 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1770 '点到点仿形：T = 0'

检测时间： 执行点到点仿形循环时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定粗加工或精加工刀具 T。

1771 ' 点到点仿形：P = 0'

检测时间：执行点到点仿形循环时。
引起原因：循环中编写的 P (点到点仿形深度) 等于 0。
解决方案：为其赋予大于 0 的值。

1772 ' 点到点仿形：空轮廓 '

检测时间：执行点到点仿形循环时。
引起原因：轮廓的前两个点 (P1 和 P2) 相同，系统认为轮廓为空。
解决方案：指定轮廓的点坐标 (从 P1 到 P12)。轮廓的最后一点为下一轮廓的第一点。

1773 ' 点到点仿形：刀具半径等于或大于切入 / 切出半径 '

检测时间：执行点到点仿形循环时。
引起原因：循环中使用的某把刀具的半径大于切入半径 (半径 P1) 或切出半径 (半径 P12) 。
解决方案：增大切入 / 切出半径或在循环中使用较小半径的刀具。

1774 ' 仿形：F = 0'

检测时间：执行仿形循环时。
引起原因：循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案：为其赋予大于 0 的值。

1775 ' 仿形：S = 0'

检测时间：执行仿形循环时。
引起原因：循环中编写的粗加工或精加工 S 等于 0。
解决方案：为其赋予大于 0 的值。

1776 ' 仿形：T = 0'

检测时间：执行仿形循环时。
引起原因：循环中未编写刀具 T。
解决方案：指定粗加工或精加工刀具 T。

1777 ' 仿形：P = 0'

检测时间：执行仿形循环时。
引起原因：循环中编写的 P (仿形深度) 等于 0。
解决方案：为其赋予大于 0 的值。

1778 ' 仿形：(精加工): 刀具的切削半径 < P'

检测时间：执行仿形循环时。
引起原因：刀具表中定义的精加工刀具的切削长度小于循环中定义的仿形深度 (P)。
解决方案：选择较大切削长度的精加工刀具。

1779 ' 狭槽铣削：F = 0'

检测时间：执行狭槽铣削循环时。
引起原因：循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案：为其赋予大于 0 的值。

1780 ' 狭槽铣削：S = 0'

检测时间：执行狭槽铣削循环时。
引起原因：循环中编写的粗加工或精加工 S 等于 0。
解决方案：为其赋予大于 0 的值。

1781 ' 狭槽铣削：T = 0'

检测时间：执行狭槽铣削循环时。
引起原因：循环中未编写刀具 T。
解决方案：指定粗加工或精加工刀具 T。



1782 ' 狭槽铣削： P = 0'

检测时间： 执行狭槽铣削循环时。
引起原因： 循环中编写的 P (仿形深度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1783 ' 狭槽铣削： L = 0'

检测时间： 执行狭槽铣削循环时。
引起原因： 循环中编写的 L (槽深) 等于 0。
解决方案： 为其赋予大于 0 的值。

1784 ' 狭槽铣削： 刀具直径小于 δ '

检测时间： 执行狭槽铣削循环时。
引起原因： 粗加工或精加工刀具的直径小于 XY 平面内的粗加工或精加工轨迹。
解决方案： 缩小 XY (δ) 方向的铣削加工轨迹，或者在机床上安装较大直径的刀具。

1785 ' 狭槽铣削： 刀具直径大于型腔 '

检测时间： 执行狭槽铣削循环时。
引起原因： 粗加工或精加工刀具得直径大于槽的尺寸 L 或 H。
解决方案： 选择较小直径的刀具加工凹槽。

1786 ' 狭槽铣削： 精加工刀具直径小于 δ '

检测时间： 执行狭槽铣削循环时。
引起原因： 精加工刀具的直径小于 XY 平面内的精加工余量 (δ)。
解决方案： 选择较小直径的刀具加工狭槽。

1787 ' 圆形型腔： F = 0'

检测时间： 执行圆形型腔循环时。
引起原因： 循环中编写的 Z 轴方向粗加工、精加工以及贯穿的进给率 F 等于 0。
解决方案： 为其赋予大于 0 的值。

1788 ' 圆形型腔： S = 0'

检测时间： 执行圆形型腔循环时。
引起原因： 循环中编写的粗加工或精加工 S 等于 0。
解决方案： 为其赋予大于 0 的值。

1789 ' 圆形型腔： T = 0'

检测时间： 执行圆形型腔循环时。
引起原因： 循环中未编写刀具 T。
解决方案： 指定粗加工或精加工刀具 T。

1790 ' 圆形型腔： P = 0'

检测时间： 执行圆形型腔循环时。
引起原因： 循环中编写的 P (型腔深度) 等于 0。
解决方案： 为其赋予大于 0 的值。

1791 ' 圆形型腔： 刀具直径小于 δ '

检测时间： 执行圆形型腔循环时。
引起原因： 粗加工或精加工刀具的直径小于 XY 平面内的粗加工或精加工轨迹。
解决方案： 缩小 XY (δ) 方向的铣削加工轨迹，或者在机床上安装较大直径的刀具。

1792 ' 圆形型腔： 刀具直径大于型腔 '

检测时间： 执行圆形型腔循环时。
引起原因： 粗加工或精加工刀具的直径大于型腔的半径 R。
解决方案： 选择直径较小的刀具加工型腔。

1793 ' 圆形型腔 : 精加工刀具直径小于 δ '

检测时间： 执行圆形型腔循环时。

引起原因： 精加工刀具的直径小于 XY 平面内的精加工余量 (δ)。

解决方案： 选择直径较大的刀具加工型腔。

1794-1800' 刀具直径 = 0'

检测时间： 执行固定循环时。

引起原因： 粗加工或精加工刀具半径为 0。

解决方案： 修正刀具表中的刀具半径值。

1801-1803' β 或 θ < 0 或者 β 或 θ > 90'

检测时间： 加工型腔时。

引起原因： 加工型腔时，编写的粗加工 (β) 或精加工 (θ) 斜穿角度不在 0º ～ 90º 之间。

解决方案： 修正 β 或 θ 值。

1804 'G87: 深度 = 0'

检测时间： 执行 G87 加工型腔时。

引起原因： 编程坐标 I 等于坐标 Z+ 距离 D。如果未编写 Z 坐标，则 I 坐标等于当前位置 + 距离 D。这意味着型腔深度为 0。

解决方案： 修正参数 I、 Z 和 / 或 D。

1805 'G87: 刀具直径大于型腔 '

检测时间： 执行 G87 加工型腔时。

引起原因： 刀具半径大于型腔的尺寸 J 和 / 或 K。

解决方案： 选择半径较小的刀具。

1806 'G87: 刀具直径小于 L'

检测时间： 执行 G87 加工型腔时。

引起原因： 刀具直径小于精加工余量 L。

解决方案： 选择较大半径的刀具。

1807 'G87: 刀具直径 = 0'

检测时间： 执行 G87 加工型腔时。

引起原因： 刀具半径等于 0。

解决方案： 修正刀具表中的刀具半径值。

1808 'G87: 刀具缺失 '

检测时间： 执行 G87 加工型腔时。

引起原因： 主轴上缺失刀具。

解决方案： 调用循环前在主轴上装载刀具。

1809 'G87: 刀具直径小于 C'

检测时间： 执行 G87 加工型腔时。

引起原因： 刀具直径小于主平面内的加工轨迹 (C) 。

解决方案： 选择较大直径的刀具或减小 C。

1810 'G88: 深度 = 0'

检测时间： 执行 G88 加工型腔时。

引起原因： 编程坐标 I 等于坐标 Z+ 距离 D。如果未编写 Z 坐标，则 I 坐标等于当前位置 + 距离 D。这意味着型腔深度为 0。

解决方案： 修正参数 I、 Z 和 / 或 D。

1811 'G88: 刀具直径大于型腔 '

检测时间： 执行 G88 加工型腔时。

引起原因： 刀具半径大于型腔的尺寸 J。

解决方案： 选择半径较小的刀具。

1812 'G88: 刀具直径小于 L'

检测时间： 执行 G88 加工型腔时。
引起原因： 刀具直径小于精加工余量 L。
解决方案： 选择较大半径的刀具。

1813 'G88: 刀具直径 = 0'

检测时间： 执行 G88 加工型腔时。
引起原因： 刀具半径等于 0。
解决方案： 修正刀具表中的刀具半径值。

1814 'G88: 刀具缺失 '

检测时间： 执行 G88 加工型腔时。
引起原因： 主轴上缺失刀具。
解决方案： 调用循环前在主轴上装载刀具。

1815 'G88: 刀具直径小于 C'

检测时间： 执行 G88 加工型腔时。
引起原因： 刀具直径小于主平面内的加工轨迹 (C)。
解决方案： 选择较大直径的刀具或减小 C。

1816 ' 参数 X 必须为参数 I 的倍数 '

检测时间： 执行多重加工循环 (G160-G165) 时。
引起原因： 参数 X 必须为参数 I 的倍数。
解决方案： 为循环中的参数赋予正确的值。

1817 ' 必须编写 I 和 K 两个参数中的一个 '

检测时间： 执行多重加工循环 (G160-G165) 时。
引起原因： 多重加工循环中未编写参数 I (加工轨迹)、 K (加工次数) 中的一个。
解决方案： 为循环中的参数赋予正确的值。

1818 'J = 0'

检测时间： 执行多重加工循环 (G160-G165) 时。
引起原因： 多重加工循环中，参数 J(加工操作之间的纵坐标轨迹) 的值为 0 。
解决方案： 为循环中的参数赋予正确的值。

1819 'K = 0'

检测时间： 执行多重加工循环 (G160-G165) 时。
引起原因： 多重循环中参数 K(加工操作次数) 的值为 0。
解决方案： 为循环中的参数赋予正确的值。

1820 ' 必须编写参数 X、 I 与 K 中的两个 '

检测时间： 执行多重加工循环 (G160-G165) 时。
引起原因： 多重加工循环中，以下参数中有两个未编程：X (长度)、 I (加工轨迹)、 K (加工操作的次数)。
解决方案： 为循环中的参数赋予正确的值。

1821 'B 必须为 I 的倍数 '

检测时间： 执行多重加工循环 (G164) 时。
引起原因： B (角行程) 必须为参数 I (加工轨迹) 的整数倍。
解决方案： 为循环中的参数赋予正确的值。

1822 ' 参数 Y 必须为参数 J 的倍数 '

检测时间： 执行多重加工循环 (G160-G165) 时。
引起原因： 参数 Y (纵坐标长度) 必须为参数 J(加工轨迹) 的整数倍。
解决方案： 为循环中的参数赋予正确的值。

1823 ' 必须编写参数 Y、 J 与 D 中的两个 '

检测时间： 执行多重加工循环 (G160-G165) 时。

引起原因： 循环 G161 和 G162 中，以下参数中有两个未编程：Y(纵坐标长度)、 J (加工轨迹)、 D (加工操作的次数)。

解决方案： 为循环中的参数赋予正确的值。

1824 ' 必须编写 I 和 K 两个参数中的一个 '

检测时间： 执行多重加工循环 (G160-G165) 时。

引起原因： 未编写两个参数 I (加工轨迹)、 K (加工操作次数) 中的一个。

解决方案： 为循环中的参数赋予正确的值。

1825 ' 必须编写参数 I 与 A 中的一个 '

检测时间： 执行多重加工循环 (G165) 时。

引起原因： 未编写两个参数 I (角程)、 A (加工操作次数) 中的一个。

解决方案： 为循环中的参数赋予正确的值。

1826 ' 参数 I 乘以 K 必须等于 360'

检测时间： 执行多重加工循环 G163 时。

引起原因： 在多重加工循环 (G163) 中，轨迹 (I) 乘以加工操作次数 (K) 不等于 360º。

解决方案： 为循环中的参数赋予正确的值。

1827 ' 参数 X 与 Y 等于 0'

检测时间： 执行多重加工循环 (G163-G165) 时。

引起原因： 定义多重加工圆心的参数的编程值为 0。

解决方案： 为循环中的参数赋予正确的值。

1828 '360 必须为参数 I 的倍数 '

检测时间： 在圆周中执行多重加工循环 (G163) 时。

引起原因： 角程 (I) 必须为 360º 的倍数。

解决方案： 为循环中的参数赋予正确的值。

1829 'I = 0'

检测时间： 执行多重加工固定循环 (G160-G165) 时。

引起原因： 参数 I (连续加工轨迹之间的距离) 的值 0。

解决方案： 为其赋予一个非 0 值。

1830 ' 刀具直径 = 0'

检测时间： 不存在。

引起原因： 用来执行循环的刀具直径为 0。

解决方案： 给刀具直径赋予一个非 0 值。

1831 'G82: C = 0'

检测时间： 执行固定循环 G82 时。

引起原因： 循环的参数 C 的编程值为 0。参数 C 指定从前一钻孔点到下一钻孔点所需要快速移动的距离 。

解决方案： 为 C 指定一个非 0 值。

1832 ' 粗加工 : I = 0'

检测时间： 执行固定循环时。

引起原因： 参数 I(最大贯穿轨迹) 的值为 0。

解决方案： 为 I 指定一个非 0 值，且小于粗加工刀具的切削长度。

1833 ' 精加工 : N = 0 且未定义刀具切削长度。

检测时间： 执行固定循环时。

引起原因： N (精加工轨迹次数) 的编程值为 0 且未定义精加工刀具的切削长度。

解决方案： 编写 N 值。



1834 ' 精加工 : Z 轴 δ 大于刀具切削长度 '

检测时间: 执行固定循环时。
引起原因: 循环参数 dz (贯穿毛坯余量) 的编程值大于精加工刀具的切削长度。
解决方案: 编写较小的 dz 或者使用另一把刀。

1835 ' 包含孤岛的型腔 : Z 轴安全位置错误 '

检测时间: 执行包含孤岛的 2D 或 3D 型腔循环。
引起原因: Z 轴安全位置 (Zs) 编写在工件内部。
解决方案: 将其编写在工件之外。

1836 ' 包含孤岛的型腔 : Z 剖面错误 '

检测时间: 执行包含孤岛的 3D 型腔循环。
引起原因: Z 轴方向编写的某个剖面不能加工。
解决方案: 确保 Z 轴方向的剖面可以加工。

1837 ' 包含孤岛的型腔 : XY 中的剖面相交 '

检测时间: 执行包含孤岛的 2D 或 3D 型腔循环。
引起原因: XY 平面中的某个剖面构成多个封闭剖面。
解决方案: 确保 XY 内的剖面只有一个公共点，且正好为其起点。

1838 ' 包含孤岛的型腔 : XY 剖面未封闭 '

检测时间: 执行包含孤岛的 2D 或 3D 型腔循环。
引起原因: XY 剖面中有一个不是封闭剖面；换句话说，它的第一点与最后一点不是同一点 。
解决方案: 确保平面内的所有剖面均封闭。

1839 ' 包含孤岛的型腔 : XY 剖面错误相交 '

检测时间: 执行包含孤岛的 2D 或 3D 型腔循环。
引起原因: 两个 XY 剖面共有某些直线或圆弧。
解决方案: 确保所有 XY 剖面只有相交点，而没有相交的直线或圆弧。

1840 ' 包含孤岛的型腔 : 内存不够，不能加工 '

检测时间: 执行包含孤岛的 2D 或 3D 型腔循环。
引起原因: 系统剩下的内存不足以继续加工型腔。
解决方案:

1841 ' 执行固定循环前刀具位置错误 '

检测时间: 执行固定循环时。
引起原因: 执行固定循环前，沿纵轴 (Z) 方向的刀尖坐标介于参考平面与工件表面之间。
解决方案: 沿纵轴 (Z) 方向将刀具移出由参考平面与工件限定的安全区域。

1842 ' 刀具直径小于 δ '

检测时间: 执行固定循环时。
引起原因: 精加工刀具直径小于精加工轨迹 (δ)。
解决方案: 选择直径大于精加工轨迹的刀具或减小精加工轨迹 (δ)。

1843 ' 包含孤岛的型腔 : 粗加工刀具半径错误 '

检测时间: 执行包含孤岛的 2D 或 3D 型腔循环。
引起原因: 粗加工操作或底部粗加工中编程刀具值为 0。或者在考虑侧面精加工余量时，刀具半径相对于型腔几何尺寸而言过大。
解决方案: 如果刀具半径不为 0，确认精加工余量的编程值是否正确。这种情况下，刀具的编程半径必须小于当前编程值。

1844 ' 包含孤岛的型腔 : 精加工刀具半径错误 '

检测时间: 执行包含孤岛的 2D 或 3D 型腔循环。
引起原因: 精加工轨迹中刀具半径编程值为 0，或者相对于型腔几何尺寸而言过大。
解决方案: 如果刀具半径不为 0，编写小于当前刀具半径的值。



1845　'G165: 弦长 (I) 大于直径 '

检测时间：　执行固定循环 G165 时。
引起原因：　参数 I 中定义的弦长大于圆周直径。
解决方案：　为参数 I 指定正确的值。

1846　' 圆弧半径为 0'

检测时间：　在圆弧中执行多重加工时。
引起原因：　在圆弧编辑器中，圆弧的编程半径为 0，或者圆心坐标 Xa 和 Ya 与所重复的循环坐标 XY 相同。
解决方案：　编写非 0 半径值；换句话说，就是为参数 Xa、 Ya 或 R 指定其他的值。

1847　' 矩形凸台： Q = 0'

检测时间：　执行矩形凸台固定循环时。
引起原因：　需要在工作平面中去除的毛坯余量使得凸台未被定义。
解决方案：　为 Q 赋予一个非 0 值。

1848　' 圆形凸台： Q = 0'

检测时间：　执行圆形凸台固定循环时。
引起原因：　需要在工作平面中去除的毛坯余量使得凸台未被定义。
解决方案：　为 Q 赋予一个非 0 值。

1849　' 矩形型腔： 精加工毛坯余量 δ 大于型腔 '

检测时间：　加工矩形型腔时。
引起原因：　不能加工所定义的型腔，因为型腔短长的一半与刀具半径的差小于工作平面内的精加工余量 (δ)。
解决方案：　减小 δ 值。

1850　'Lx 必须为 Ix 的倍数 '

检测时间：　执行 G160 和 G161 循环时。
引起原因：　直线或矩形的长度必须为加工操作之间的距离 Ix 的倍数。否则，加工操作将不是等距离的。
解决方案：　修正 Lx 和 Ix 的值。

1851　'Ly 必须为 Iy 的倍数 '

检测时间：　执行 G160 和 G161 循环时。
引起原因：　直线或矩形的长度必须为加工操作之间的距离 Iy 的倍数。否则，加工操作将不是等距离的。
解决方案：　修正 Ly 和 Iy 的值。

1852　' 未对刀具进行编程 '

检测时间：　执行校准循环 (#PROBE1/2) 或测量循环 (#PROBE 3-9) 时。
引起原因：　执行循环前，主轴上未装载刀具。
解决方案：　执行循环时，主轴上需要装载刀具 (#PROBE1) 或探头 (#PROBE 2-9)。

1853　' 未接收到探头信号 '

检测时间：　执行 #PROBE 循环时。
引起原因：　执行 #PROBE 循环时，在检测运动期间未检测到任何点。
解决方案：　修正循环的几何参数，使探头能真正接触到刀具(#PROBE1)或工件(#PROBE 2-9)。

1854　' 直径 J 必须大于 0 '

检测时间：　执行 #PROBE 2 或 #PROBE 9 循环时。
引起原因：　直径的编程值为负。
解决方案：　指定一个正值。

1855　' 刀具直径大于孔的直径 '

检测时间：　执行 #PROBE 2 或 #PROBE 9 循环时。
引起原因：　待测量的孔直径或用来校准探头的孔直径小于测量中使用的探头的直径。
解决方案：　赋予正确的值。



1856 ' 回撤距离 E 错误 '

检测时间： 执行 #PROBE 2 或 #PROBE 9 循环时。
引起原因： 回撤距离不能为负且不能大于孔直径。
解决方案： 为 E 指定一个介于 0 到孔直径之间的值。

1857 ' 未选择刀具偏置 '

检测时间： 执行校准循环时 (#PROBE1/2)。
引起原因： 执行刀具校准循环或检测循环前未加载刀具偏置 (D)。
解决方案： 在调用循环前编写刀具偏置 D。

1859 ' 探头校准的有效平面必须为 G17、 G18 或 G19'

检测时间： 执行 #PROBE 2 循环时。
引起原因： 开始探头校准循环时， G20 有效。
解决方案： 在调用循环前编写 G17 (XY 平面)、 G18 (ZX 平面) 或 G19 (YZ 平面)。

1860 ' 编程：K = 0/1/2'

检测时间： 执行 #PROBE 3 循环 (表面测量循环) 时。
引起原因： 参数 K 的值只能为 0、 1 或 2。
解决方案： 参数 K 定义测量表面所用的轴。可为 K 指定下述值：

- 0：横坐标轴。
- 1：纵坐标轴。
- 2：纵轴。

1861 ' 安全距离 B 必须大于 0'

检测时间： 执行校准或测量循环时 (#PROBE)。
引起原因： 参数 B 的编程值为负。
解决方案： 赋正值。

1862 '#PROBE 1: G20 有效时不允许使用 '

检测时间： 执行 #PROBE 1 循环时。
引起原因： 开始探头校准循环时， G20 有效。
解决方案： 在调用循环前编写 G17 (XY 平面)、 G18 (ZX 平面) 或 G19 (YZ 平面)。

1863 ' 刀具磨损 '

检测时间： 执行刀具校准循环 #PROBE1 时。
引起原因： 循环检测到磨损刀具。被检测刀具的磨损量大于许可范围 (参数 L 或 M)。
解决方案： 更换刀具并重新执行循环。

1864 '#PROBE 1: 参数 U、 V、 W 必须大于 X、 Y、 Z'

检测时间： 执行刀具校准循环 #PROBE1 时。
引起原因： 循环中，定义探头位置的坐标错误。UVW 为上坐标，XYZ 为下坐标。
解决方案： 为各参数赋值，使 U > X、 V > Y、 W > Z。

1865 ' 包含孤岛的型腔 : 粗加工 T 缺失 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 粗加工操作中编写的刀具号 (T) 在刀具表中不存在。
解决方案： 将该刀具插入刀具表或从刀具表中编写另一个刀具号。

1866 ' 包含孤岛的型腔 : 粗加工进给率 F 错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 粗加工操作的加工进给率 (F) 的编程值为 0。
解决方案： 为 F 编写一个非 0 值。

1867 ' 包含孤岛的型腔 : 粗加工转速 S 错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 粗加工操作的主轴转速 (S) 的编程值为 0。
解决方案： 为 S 编写一个非 0 值。



1868 ' 包含孤岛的型腔： 精加工 T 缺失 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 精加工操作中编写的刀具号 (T) 在刀具表中不存在。
解决方案： 将该刀具插入刀具表或从刀具表中编写另一个刀具号。

1869 ' 包含孤岛的型腔： 精加工进给率 F 错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 精加工操作的加工进给率 (F) 的编程值为 0。
解决方案： 为 F 编写一个非 0 值。

1870 ' 包含孤岛的型腔： 精加工转速 S 错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 精加工操作的主轴转速 (S) 的编程值为 0。
解决方案： 为 S 编写一个非 0 值。

1871 ' 包含孤岛的型腔： 粗轧孔型错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 粗加工操作中编写的粗轧孔型 (D) 大于刀具直径。
解决方案： 粗轧孔型的编程值小于或等于刀具直径。

1872 ' 包含孤岛的型腔： 精加工轨迹错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 精加工操作中编写的 2D 型腔底部加工轨迹 (D) 大于刀具直径。或者 3D 型腔的精加工轨迹 (e) 编程值为 0。
解决方案： 在两种情形下分别编写正确的值。

1873 ' 包含孤岛的型腔： 侧面毛坯余量 (呆滞材料) 错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 精加工操作侧面毛坯余量 (d) 大于操作中使用的刀具直径。
解决方案： 侧面毛坯余量的编程值小于或等于刀具直径。

1874 ' 包含孤岛的型腔： 深度错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 型腔深度 (P) 的编程值为 0。
解决方案： 编写非 0 值。

1875 ' 包含孤岛的型腔： XY 剖面不存在 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 编写的 XY 剖面不存在。
解决方案： 更改剖面名或生成该剖面。

1876 ' 包含孤岛的型腔： XY 剖面错误 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： XY 剖面的几何尺寸编程错误。
解决方案： 进入剖面编辑器，编辑剖面，并检查几何尺寸的哪一部分编程错误。

1877 ' 包含孤岛的型腔： Z 剖面不存在 '

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。
引起原因： 编写的 Z 剖面不存在。
解决方案： 更改剖面名或生成该剖面。

1878 ' 包含孤岛的型腔： 精加工刀尖半径错误 '

检测时间： 执行包含孤岛的 3D 型腔循环时。
引起原因： 精加工刀具的刀尖半径 (Rp) 大于其半径 (R)。
解决方案： 刀尖半径编程值小于或等于半径。

1879 '包含孤岛的型腔：半精加工刀具半径错误'

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。

引起原因： 可能的原因包括：
- 2D 型腔底部的精加工刀具半径为 0，或者相对于型腔几何尺寸而言过大。
- 3D 型腔的半精加工刀具半径为 0。

解决方案： 两种情形下，分别编写正确的值。

1880 '包含孤岛的型腔：半精加工 T 缺失'

检测时间： 执行包含孤岛的 3D 型腔循环时。

引起原因： 半精加工操作中编写的刀具号 (T) 在刀具表中不存在。

解决方案： 将该刀具插入刀具表或从刀具表中编写另一个刀具号。

1881 '包含孤岛的型腔：半精加工进给率 F 错误'

检测时间： 执行包含孤岛的 3D 型腔循环时。

引起原因： 半精加工操作的加工进给率 (F) 的编程值为 0。

解决方案： 为 F 编写一个非 0 值。

1882 '包含孤岛的型腔：半精加工转速 S 错误'

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。

引起原因： 半精加工操作的主轴转速 (S) 的编程值为 0。

解决方案： 为 S 编写一个非 0 值。

1883 '包含孤岛的型腔：正在生成型腔'

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。

引起原因： 不能模拟 2D 或 3D 型腔，因为正在生成或模拟型腔。

解决方案： 等待其结束，然后重新执行。

1884 '包含孤岛的型腔：剖面中的圆弧编程错误'

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。

引起原因： 构成型腔几何尺寸的某个剖面的圆弧编程错误。原因可能为：
- 未在循环中编辑。
- 剖面已经被程序而不是剖面编辑器修改。
- 剖面轴的机床参数 DIAMPROG 设置不正确。

解决方案： 进入循环界面，编辑型腔的每个剖面，找出编程错误的圆弧。一旦找到错误的剖面与圆弧，进入编辑器对圆弧进行修改。

1885 '包含孤岛的型腔：型腔几何尺寸错误'

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环时。

引起原因： 构成型腔的某些剖面的某些轨迹编程错误 (圆角、倒角等)。

解决方案： 进入循环界面，编辑型腔的每个剖面并找出错误。一旦找到错误以及所在剖面，进入编辑器对其进行修改。

1886 '包含孤岛的型腔：粗加工贯穿角错误'

检测时间： 执行包含孤岛的 2D 或 3D 型腔循环。

引起原因： 粗加工刀具贯穿角 (Ae) 编程值小于或等于 0º，或大于 90º。

解决方案： 编写大于 0º 且小于 90º 的值。

1887 '包含孤岛的型腔：半精加工刀具贯穿角错误'

检测时间： 执行包含孤岛的 3D 型腔时。

引起原因： 半精加工刀具贯穿角 (Ae) 编程值小于或等于 0º，或大于 90º。

解决方案： 编写大于 0º 且小于 90º 的值。





2000-2999 号错误

2000 '刀具半径大于圆弧半径'

检测时间：　执行程序时。
引起原因：　刀具半径大于待加工圆弧半径。
解决方案：　使用较小半径的刀具。

2001 '刀具半径补偿破坏剖面'

检测时间：　执行程序时。
引起原因：　相对于待加工剖面而言，刀具半径过大，这样会破坏剖面。
解决方案：　使用较小半径的刀具。

2002 '线性补偿的第一个程序段必须为线性的'

检测时间：　执行程序时。
引起原因：　激活刀具半径补偿 (G41 或 G42) 后，下一运动程序段为圆周程序段。不能在圆周程序段中开始刀具半径补偿。
解决方案：　刀具半径补偿必须在线性程序段中开始。因此，G41-G42 后的运动程序段必须为线性运动程序段。

2003 '加工连续圆弧的刀具半径过大'

检测时间：　执行程序时。
引起原因：　加工构成环（彼此相交）的两个连续圆弧时，刀具半径过大，而不能加工环的内侧。
解决方案：　使用较小半径的刀具。

2004 '含有刀具半径补偿的程序段间有太多非运动程序段'

检测时间：　执行程序时。
引起原因：　刀具半径补偿有效时，运动程序段间编写了太多非运动程序段（参数指定 P、变量等）。
解决方案：　减小非运动程序段数目。可将几个非运动程序段并为一个程序段。

2005 '线性补偿的最后一个程序必须为线性的'

检测时间：　执行程序时。
引起原因：　取消刀具半径补偿 (G40) 后，下一程序段为圆周程序段。刀具半径补偿不能在圆周程序段中结束。
解决方案：　刀具半径补偿必须在线性程序段中结束。因此， G40 后的运动程序段必须为线性运动程序段。

2006 '必须在线性轨迹上改变刀具半径补偿 (G41/G42)'

检测时间：　执行程序时。
引起原因：　刀具半径补偿被更改，且下一运动程序段为圆周程序段。换句话说， G41 有效时编写了 G42，或在 G42 有效时编写了 G41。如果下一运动程序段为圆弧，则不能更改刀具半径补偿。
解决方案：　在 (G41-G42) 后编写线性运动程序段。

2007 'G138 有效时，第一个补偿程序段后不允许使用 G40'

检测时间：　执行程序时。
引起原因：　第一补偿程序段由直接选择 (G138) 处理时，编写了刀具半径补偿 (G40)。
解决方案：　为了能够开始补偿并取消它，使用直接补偿选择 (G138) 时，需要一个平面内的附加运动程序段。也可以使用间接补偿选择 (G139)，它不需要附加程序段。

2008 'G138 有效时，第一补偿程序段后不允许进行 G41/G42 变换 '

检测时间： 执行程序时。

引起原因： 处理直接选择 (G138) 的第一补偿程序段时，编写了从 G41 到 G42 或从 G42 到 G41 的变换。

解决方案： 为了能够开始补偿并取消它，使用直接补偿选择 (G138) 时，需要一个平面内的附加运动程序段。可在该运动后变换补偿类型。也可以使用间接补偿选择 (G139)，它不需要附加程序段。

2010 ' 计算圆弧间的刀具半径时，刀具半径过大 '

检测时间： 执行程序时。

引起原因： 加工圆弧的刀具半径过大。

解决方案： 可采用的解决办法有：

- 使用较小的刀具半径加工圆弧。
- 从编程剖面中删除该圆弧。
- 编写较大的圆弧。

2011 ' 圆周轨迹的补偿运动 (剖面被破坏)'

检测时间： 执行程序时。

引起原因： 加工圆弧的刀具半径过大。

解决方案： 可采用的解决办法有：

- 使用较小的刀具半径加工圆弧。
- 从编程剖面中删除该圆弧。
- 编写较大的圆弧。

2013 ' 前一轨迹与圆弧间的刀具半径不一致 '

检测时间： 执行程序时。

引起原因： 导致该错误的原因是：补偿圆弧时，刀具半径相对于前一程序段发生变化。

解决方案： 补偿圆弧时，不要改变刀具半径。

2016 ' 为避免冲突，改变剖面 '

检测时间： 执行程序时。

引起原因： 冲突检测或刀具补偿过程中，检测到某个刀具运动可能会破坏剖面，因此将其删除。

解决方案： 根据检测到的冲突，有以下几个解决办法：使用较小半径的刀具、以不同的方法编写剖面或接受冲突检测程序给出的改变。

2017 ' 冲突检测有效时不能进行编程 '

检测时间： 执行程序时。

引起原因： 进行冲突检测时，不允许进行某些操作，比如原点搜索、探测等。

解决方案： 如果编写了上述任何一个操作，就不要激活冲突检测。

2100 ' 超出 G5-G60 的正软件极限 '

检测时间： 执行程序时。

引起原因： CNC 生成的圆角路径超出有效行程。

解决方案： 不使用 G50 或 G7 圆角编程。

2101 ' 超出 G5-G60 的负软件极限 '

检测时间： 执行程序时。

引起原因： CNC 生成的圆角路径超出有效行程。

解决方案： 不使用 G50 或 G7 圆角编程。

2102 '#ROUNDPAR 指令的参数过小 '

检测时间： 执行程序时。

引起原因： 参数过小，不能进行处理。

解决方案： 该指令的编程值应大于 20 微米。

2103 '#ROUNDPAR 指令的参数类型错误 '

检测时间：执行程序时。
引起原因：圆角类型编程错误。
解决方案：根据手册，编写有效值。

2106 'HSC 模式内部错误 '

检测时间：执行程序时。
引起原因：计算 HSC 模式中的工作路径时出错。
解决方案：在程序段中取消导致出错的 HSC 模式。

2108 ' 不允许改变 HSC 模式 '

检测时间：执行程序时。
引起原因：试图在 HSC 模式有效时编写另一个 HSC 模式。
解决方案：在编写另一个 HSC 模式前取消当前的 HSC 模式。

2109 'HSC: 轮廓错误——太小 '

检测时间：执行程序时。
引起原因：轮廓太小，不能进行处理。
解决方案：该指令的编程值应大于 20 微米。

2110 'HSC 模式中超出正软件极限 '

检测时间：执行程序时。
引起原因：CNC 生成的 HSC 模式工作路径超出有效行程。
解决方案：在程序段中取消导致出错的 HSC 模式。

2111 ' 模式中超出负软件极限 '

检测时间：执行程序时。
引起原因：CNC 生成的 HSC 模式工作路径超出有效行程。
解决方案：在程序段中取消导致出错的 HSC 模式。

2112 ' 样条的初始切线未确定 '

检测时间：执行程序时。
引起原因：没有一个起点与前面的点相同，因此切线未确定或只编写了切线的一根轴。
解决方案：为样条编写一个预切入点。

2113 ' 样条的结束切线未确定 '

检测时间：执行程序时。
引起原因：没有一个终点与前面的点相同，因此切线未确定或只编写了切线的一根轴。
解决方案：为样条编写一个切出点。

2114 ' 不允许使用样条激活指令 '

检测时间：执行程序时。
引起原因：编写了样条模式激活指令，但是前一个样条并未结束。
解决方案：定义新的样条前先取消当前样条。

2115 '#ASPLINE 指令参数错误 '

检测时间：执行程序时。
引起原因：切线类型错误。
解决方案：根据手册中指定的类型编程。

2116 ' 激活样条出错 '

检测时间：执行程序时。
引起原因：激活样条模式时未取消前一模式。
解决方案：在激活该样条模式前，先取消当前模式。



2118 '样条：样条类型错误'

检测时间： 执行程序时。
引起原因： 不允许使用该样条类型。
解决方案： 根据手册中指定的样条类型进行编程。

2119 '样条：样条有效时不能更改类型'

检测时间： 执行程序时。
引起原因： 试图更改样条类型，但是前一类型仍然有效。
解决方案： 先取消当前样条，然后再编写另一类型的样条。

2121 '样条有效时不允许编写圆周轨迹 (圆弧) '

检测时间： 执行程序时。
引起原因： 样条有效时编写了圆弧。
解决方案： 先取消样条模式，然后再编写圆弧。

2122 '样条有效时不允许编写多项式程序段'

检测时间： 执行程序时。
引起原因： 样条有效时编写了多项式运动。
解决方案： 先取消样条模式，然后再编写多项式。

2123 '样条有效时不允许编程'

检测时间： 执行程序时。
引起原因： 样条有效时试图改变坐标系统。
解决方案： 先取消样条模式，然后再更换坐标系统。

2124 '样条至少需要两根主轴'

检测时间： 执行程序时。
引起原因： 编写样条指令时通道中只有一根轴。
解决方案： 不在样条模式中编写运动。

2125 '样条：切线计算错误'

检测时间： 执行程序时。
引起原因： 某个点重复编程。
解决方案： 编写不同的点。

2126 '超出样条的正软件极限'

检测时间： 执行程序时。
引起原因： CNC 生成的 SPLINE 模式工作路径超出有效行程。
解决方案： 在程序段中取消导致出错的 SPLINE 模式。

2127 '超出样条的负软件极限'

检测时间： 执行程序时。
引起原因： CNC 生成的 SPLINE 模式工作路径超出有效行程。
解决方案： 在程序段中取消导致出错的 SPLINE 模式。

2128 '螺旋轨迹编程错误'

检测时间： 执行程序时。
引起原因： 螺旋轨迹的一个参数缺失或错误。编写了螺距，但未编写深度；或者缺失螺距与终点；或者深度、螺距与终点彼此不谐调。
解决方案： 编写能与螺距和深度谐调的终点。如果为转体，则编写螺距与深度。

2129 'RTCP 转换错误'

检测时间： 执行程序时。
引起原因： 编写了 RTCP，但是通道中没有该转换所需的所有轴。
解决方案： 激活 RTCP 前，为通道配置所有必需的轴。



2130 ' 斜面转换错误 '

检测时间： 执行程序时。
引起原因： 编写了斜面，但是通道中没有三根轴。该转换需要三根轴。
解决方案： 激活斜面前，为通道配置所有必需的轴。

2131 'C 轴模式错误 '

检测时间： 执行程序时。
引起原因： 程序段中不能进行 C 轴转换。
解决方案： 删除导致出错的程序段。将信息发送至 FAGOR。

2133 ' 主轴缺失 '

检测时间： 执行程序时。
引起原因： 耦合的主轴在通道中不可用。
解决方案： 利用通道中的轴编写耦合。

2134 ' 从轴缺失 '

检测时间： 执行程序时。
引起原因： 耦合的从轴在通道中不可用。
解决方案： 利用通道中的轴编写耦合。

2135 ' 编写的最大主轴转速为 0'

检测时间： 执行程序时。
引起原因： 恒表面转速模式中，最大主轴转速的编程值为 0。
解决方案： 编写主轴最大 rpm（转 / 分钟）。

2136 ' 编写的最大切削速度为 0'

检测时间： 执行程序时。
引起原因： 恒表面转速模式中，未编写切削速度。
解决方案： 编写切削速度。

2137 ' 未定义 FACEAXIS（曲面轴）'

检测时间： 执行程序时。
引起原因： 未定义曲面轴。
解决方案： 设置曲面轴的机床参数 FACEAXIS = TRUE。

2138 ' 螺纹加工轴缺失 '

检测时间： 执行程序时。
引起原因： 螺纹加工轴不在通道中。
解决方案： 如果所有轴都不可用，则不要编写螺纹加工操作，或者配置通道轴以加工螺纹。

2139 ' 未编写螺纹加工运动 '

检测时间： 执行程序时。
引起原因： 螺纹加工轴未编写任何运动。
解决方案： 为螺纹加工轴编写运动。

2140 '#SLOPE 指令参数错误 '

检测时间： 执行程序时。
引起原因： #SLOPE 指令的某个参数编程错误。
解决方案： 查阅手册，编写正确值：
#SLOPE [类型 (0...2)、加加速度 (0...2)、加速度 (0,2)、运动 (0,1)]

2141 ' 不允许编写圆周轨迹 (少于 2 根轴)'

检测时间： 执行程序时。
引起原因： 如果通道中少于 2 根轴，则不能编写圆周轨迹。
解决方案： 为通道配置至少两根轴，进行圆周插补。



2142 ' 不允许与 Hirth 轴编程 '

检测时间：	执行程序时。
引起原因：	编写了与 Hirth 轴不相容的指令。
解决方案：	核对程序。

2143 ' 不能计算样条的切线 '

检测时间：	执行程序时。
引起原因：	不能计算未选择的样条的切线。
解决方案：	释放另一个点或为未经选择的样条编写另一条切线。

2144 ' 生成样条时出错 '

检测时间：	执行程序时。
引起原因：	激活正切样条时未编写第一个程序段。
解决方案：	激活正切样条前先进行运动编程。

2145 ' 结束程序时未取消样条 '

检测时间：	执行程序时。
引起原因：	到达程序末尾时样条模式仍然有效。
解决方案：	结束程序前，先编写 SPLINE OFF 指令。

2146 ' 不能取消样条 '

检测时间：	执行程序时。
引起原因：	不能读取足够的程序段来取消样条。
解决方案：	只有一个程序段时不要编写样条。

2147 'HSC 模式中不能编写样条 '

检测时间：	执行程序时。
引起原因：	HSC 模式有效时编写了 SPLINE ON 指令。
解决方案：	首先使用 #HSC OFF 指令取消 HSC 模式，或者不要编写样条模式。

2148 ' 样条模式中，程序段之间有太多非运动程序段 '

检测时间：	执行程序时。
引起原因：	运动程序段离得太远。
解决方案：	尽可能将运动程序段之间的几个程序段并为一个程序段，从而对其进行压缩，或者先取消样条，然后再重新激活。

3000-3999 号错误

3000 ' 圆弧不能与从轴编写在一起 '

检测时间： 执行程序时。
引起原因： 可能的原因包括：
1. 单独移动耦合中的某根轴。
2. 单独移动龙门对中的某根轴。
解决方案： 不能单独移动从轴。要移动从轴，必须移动与其关联的主轴。

3001 ' 不能在 DRO 模式中移动轴 '

检测时间： 执行程序时。
引起原因： 试图移动处于 DRO 模式中的轴。
解决方案： 不要试图移动 DRO 轴或将其设置为非 DRO 轴。

3002 ' 超出正软件行程 '

检测时间： 重定位轴时。
引起原因： 将轴重新定位于某个软件行程之外的点。不可能到达该点，以继续执行程序。
解决方案： 确保轴位于软件行程之内。

3003 ' 超出负软件行程 '

检测时间： 重定位轴时。
引起原因： 将轴重新定位于某个软件行程之外的点。不可能到达该点，以继续执行程序。
解决方案： 确保轴位于软件行程之内。

3005 ' 初始化探测程序时位置控制出错 '

检测时间： 初始化探测程序时。
引起原因： 这是所有编程轴中探测程序的安全功能。编程轴无效或不可用。
解决方案： 确保编程轴有效且可用。

3007 ' 运动开始前未接收到探测信号 '

检测时间： 检测探头信号后。
引起原因： 激活探测程序前未编写 G100 功能。
解决方案： 请与厂商联系。

3008 ' 探测程序出错 '

检测时间： 检测探头信号后。
引起原因： 这是所有编程轴中探测程序的安全功能。
解决方案： 确保编程轴有效且可用。

3010 ' 轴未定义为探头 (PROBEAXIS)'

检测时间： 执行程序时。
引起原因： 试图使用未包含在探测运动中的轴进行探测。
解决方案： 改变轴的参数 PROBEAXIS。



3011 ' 编写的探头轴过多 '

检测时间： 执行探测程序段时。
引起原因： 探测程序中编写的轴数量超出通道中轴的最大数量。
解决方案： 修改探测程序段。

3013 ' 刹车距离大于参数 PROBERANGE'

检测时间： 执行程序时。
引起原因： 当前速度所需的刹车距离大于参数 PROBERANGE。
解决方案： 增加刹车距离 (PROBERANGE)，或减小探测进给率。

3015 ' 程序段搜索期间，遗漏了原点搜索 '

检测时间： 执行程序时。
引起原因： 试图在执行空运行时搜索原点。
解决方案： 在程序外搜索原点。

3016 ' 程序段搜索期间，遗漏了手动模式 '

检测时间： 执行程序时。
引起原因： 程序段搜索期间，接受了手动模式请求。
解决方案： 不要在程序段搜索期间发送手动模式请求，同时记住手动模式无效的警告。

3017 ' 程序段搜索已经激活 '

检测时间： 程序段搜索期间。
引起原因： 试图激活已经被激活的程序段搜索。
解决方案： 请与厂商联系。

3018 ' 尚未达到恒表面速度 '

检测时间： 执行程序时。
引起原因： 主轴转速未达到编程值。
解决方案： 解决办法有：

- 减小恒表面速度 S。
- 减小轴进给率。
- 先编写 S，然后留出足够时间使主轴能达到该速度。

3019 'G95 中编写的进给率太低 '

检测时间： 执行程序时。
引起原因： 进给率太低。
解决方案： 增加进给率或增加主轴 rpm（转 / 分钟）。

3020 ' G96 中未编写 S'

检测时间： 执行程序时。
引起原因： 主轴转速为 0。
解决方案： 编写转速。

3021 ' 最大恒表面速度为 0'

检测时间： 执行程序时。
引起原因： 最大主轴转速为 0。
解决方案： 编写转速。

3022 ' 初始化轴坐标时出错 '

检测时间： 执行程序时。
引起原因： 轴坐标不匹配。
解决方案： 联系 Fagor 经销商。

3023 ' 新坐标系统中缺失轴 '

检测时间： 执行程序时。
引起原因： 定义新的坐标系统时，通道中三根轴的某根未激活。
解决方案： 使用 #SET AX[ ] 指令激活通道中所需的三根轴。



3025 ' 超出正软件行程 '

检测时间： 执行程序时。
引起原因： 试图将轴移动至软件行程以外的某个坐标。该行程可以通过程序中的机床参数或实时设置。
解决方案： 确保轴位于软件行程之内。

3026 ' 超出负软件行程 '

检测时间： 执行程序时。
引起原因： 试图将轴移动至软件行程以外的某个坐标。该行程可以通过程序中的机床参数或实时设置。
解决方案： 确保轴位于软件行程之内。

3027 ' 计算反向 RTCP 转换时出错 '

检测时间： 执行程序时。
引起原因： 不能执行从工件坐标到机床坐标的转换。
解决方案： 联系 Fagor 经销商。

3029 ' 计算直接 RTCP 转换时出错 '

检测时间： 执行程序时。
引起原因： 不能执行从机床坐标到工件坐标的转换。
解决方案： 联系 Fagor 经销商。

3030 ' 模拟轴不能与非模拟轴混合 '

检测时间： 执行程序时。
引起原因： 使用探头进行测量时，模拟轴与非模拟轴混合。
解决方案： 使用探头进行测量时，使用相同类型的轴。

3031 ' 当前状态下不允许使用 (RTCP/TLC) 补偿 '

检测时间： 执行程序时。
引起原因： 另一个补偿有效时，试图激活 RTCP 或 TLC 补偿。
解决方案： 试图激活所需的转换前，先执行 #RTCP OFF 或 TLC OFF 指令。

3032 'DRO 模式中不允许进行原点搜索 '

检测时间： 执行程序时。
引起原因： 试图回零 DRO 轴。
解决方案： 不要回零 DRO 轴。

3033 ' 轴位于旋转中心时不能切换至手动模式 '

检测时间： 执行程序时。
引起原因： 曲面轴 C 转换有效且轴位于旋转中心时，试图激活手动模式。
解决方案： 在旋转中心以外的位置激活手动模式。

3034 ' 组号错误 '

检测时间： 执行程序时。
引起原因： 访问一组不存在的轴参数。
解决方案： 组号必须介于 1 ～ NPARSET 之间。

3035 ' 从轴跟随误差的差过大 '

检测时间： 执行程序时。
引起原因： 龙门轴的主轴与从轴之间的跟随误差大于 MAXCOUPLE 或耦合轴的编程误差值。
解决方案： 调整耦合轴的动态性能或增加两根轴跟随误差的最大许可差值。

3036 ' 换档错误 '

检测时间： 执行程序时。
引起原因： CNC 历史记录中的主轴齿轮与 PLC 中显示的待激活主轴齿轮不一样。
解决方案： 检查 PLC 程序。

3037 ' 轴锁定 '

检测时间： 执行程序时。
引起原因： 试图移动轴，但是 SERVO 没有打开（ON）。
解决方案： 检查 PLC 程序中 SERVO 信号的处理，或者增加轴的 DWELL（暂停）参数值 。

3038 ' 等待报告的参数过多 '

检测时间： 执行程序时。
引起原因： 系统过载。
解决方案： 关闭与 CNC 无关的 WINDOWS 应用程序。联系 Fagor 公司。

3039 ' 程序段搜索过程中未发现停止程序段 '

检测时间： 执行空运行时。
引起原因： 以空运行模式执行程序时，未找到停止程序段。
解决方案： 改变停止程序段。

3040 'Hirth 轴定位错误 '

检测时间： 执行程序时。
引起原因： Hirth 轴未定位在其步距的整倍号处。
解决方案： 将 Hirth 轴定位在有效位置，或取消该 Hirth 轴。

3041 'G33/G95 未定义主轴 '

检测时间： 执行程序时。
引起原因： G33 和 G95 需要主轴。该主轴缺省状态下为通道的主轴，但是如果 PLC 寄存器 SYNC 指定另一根主轴，则会使用后者进行工作。
如果在某个特定时刻，两根轴都未定义，则会出错。
解决方案： 在 PLC 寄存器中指定一根轴，或在通道中定义一根主轴。

3043 ' 激活自动螺纹加工时，主轴位于 M5 中 '

检测时间： 执行程序时。
引起原因： 缺省状态下，自动螺纹加工 (G33) 中的主轴位于 M5 中，或者 M5 与 G33 位于同一程序段中。
解决方案： 首先编写主轴旋转，或在 G33 所在程序段中指定旋转方向 (M3/M4) 或转速 (S)。

3500 ' 第 1 部分的线加速度小于或等于 0 '

检测时间： 启动或执行程序时。
引起原因： 编写的加速度为 0。
解决方案： 编写正的加速度值。

3501 ' 第 1 部分的线加速度大于最大值 '

检测时间： 启动或执行程序时。
引起原因： 编写的加速度大于最大值。
解决方案： 编写小于最大值的加速度值。

3502 ' 第 2 部分的线加速度小于或等于 0'

检测时间： 启动或执行程序时。
引起原因： 编写的加速度为 0。
解决方案： 编写正的加速度值。

3503 ' 第 2 部分的线加速度大于最大值 '

检测时间： 启动或执行程序时。
引起原因： 编写的加速度大于最大值。
解决方案： 编写小于最大值的加速度值。

3504 ' 加速度变化速度大于最大值 '

检测时间： 启动或执行程序时。
引起原因： 编写的加速度变化速度大于最大值。
解决方案： 编写小于最大值的变化速度。

3505 '加加速度极限过冲'

检测时间: 执行程序时。
引起原因: 加加速度在轨迹上过冲。
解决方案: 联系 Fagor 经销商。

3506 '超过加加速度极限'

检测时间: 执行频率分析指令时。
引起原因: 振幅的频率过高。
解决方案: 减小最大频率或降低振幅。

3507 '超过加速度极限'

检测时间: 执行程序时。
引起原因: 轨迹上的加速度过大。
解决方案: 联系 Fagor 经销商。

3508 '速度的频率过高'

检测时间: 执行频率分析指令时。
引起原因: 附加速度低于最大频率速度。
解决方案: 减小最大频率或编写较高的附加速度。

3600 '主轴转速为 0'

检测时间: 执行程序时。
引起原因: 未编写主轴定位转速。
解决方案: 核对程序，确保有效主轴齿轮的参数 REFFEED2 和 G00FEED 不为 0。

3601 '主轴转速大于最大极限'

检测时间: 执行程序时。
引起原因: 主轴转速大于机床参数为当前齿轮设置的最大值。
解决方案: 检查主轴参数 G00FEED。

3602 'DRO 模式中不能移动主轴'

检测时间: 执行程序时。
引起原因: 通过参数 PLC. DROS = 1 和 SERVOSON = 0，使主轴位于 DRO 模式。
解决方案: 检查 PLC 程序中的标志 DRO 和 SERVO。

3603 '主轴定位超出模块范围'

检测时间: 执行程序时。
引起原因: 将主轴定位在由机床参数 MODUPLIM 和 MODLOWLIM 设定的范围之外的某个坐标。
解决方案: 检查参数 MODUPLIM 和 MODLOWLIM。

3604 '主轴定位方向与机床参数设置的方向相反'

检测时间: 执行程序时。
引起原因: 主轴定位需要主轴在机床参数 UNIDIR 设置的相反方向运动。
解决方案: 检查主轴机床参数 UNIDIR。

3605 '主轴旋转方向与机床参数设置的方向相反'

检测时间: 执行程序时。
引起原因: 主轴旋转的方向与机床参数 UNIDIR 设置的方向相反。
解决方案: 检查主轴机床参数 UNIDIR。

3606 '主轴定位需要绝对坐标'

检测时间: 执行程序时。
引起原因: 主轴开环旋转后，必须以绝对坐标进行定位。
解决方案: 以绝对坐标进行编程。

3700 '超出轴行程'

检测时间： 轴运动期间。
引起原因： 超出轴行程 (PLC)。
解决方案： 核对程序。

3701 '参考位置位于软件极限之外'

检测时间： 确认机床参数时。
引起原因： 参数 REFVALUE 的值超出轴软件极限范围。
解决方案： 改变参数。

3702 '轴跟随误差超出极限'

检测时间： 执行程序时。
引起原因： 超出由机床参数设置的跟随误差许可值。可能的原因：轴调节器错误，某些使能缺失，电机、驱动器、反馈系统和 / 或机械故障。
解决方案： 检查参数，调节器，线路连接以及驱动器、反馈系统以及电机的状态。

3703 '超出正软件行程'

检测时间： 轴运动期间。
引起原因： 超出机床参数 LIMITPOS 或变量 RTLIMITPOS 设定的坐标。
解决方案： 核对程序。

3704 '超出负软件行程'

检测时间： 轴运动期间。
引起原因： 超出机床参数 LIMITPOS 或变量 RTLIMITPOS 设定的坐标。
解决方案： 核对程序。

3705 '需要原点搜索'

检测时间： 定位未回零的主轴时。
引起原因： 由开环切换至闭环时丢失主轴参考点 (原点) 。
解决方案： 请与厂商联系。

3706 '与探头相关的操作出错'

检测时间： 初始化探测程序时出错。
引起原因： 与探头相关的数字输入无效。
解决方案： 检查探头参数。

3707 '搜索原点错误'

检测时间： 模拟轴与 SERCOS 轴搜索过程中。
引起原因： 原点搜索失败时，CAN 计数器与 SERCOS 驱动器报告错误。可能由参数设置错误，计数器 、 SERCOS 驱动器、反馈系统以及线路连接故障等引起。
解决方案： 检查与原点搜索相关的参数以及计数器模块或 SERCOS 驱动器、反馈系统、线路连接等的状态。

3708 '超过进入适当区域的最大时间'

检测时间： 轴运动期间。
引起原因： 到达适当位置的时间大于参数 INPOMAX 设定的值。
解决方案： 调整参数 INPOMAX。

3709 '更新模拟输入时出错'

检测时间： 读取模拟输入时。
引起原因： 循环读取模拟输入失败。这可能由 COMPCI、 CAN 总线、模拟输入模块等中的问题引起。
解决方案： 检查 CAN 总线、模拟输入模块、线路连接等的状态。

3710 '定位指令值超出规定范围'

检测时间： 执行程序时。
引起原因： 定位指令变量超值。超出了 SGN32 的最大值。
解决方案： 检查轴的参数设置 (增益、ServoOff、分辨率等) 。



3711 ' 更新 CAN 反馈读数时出错 '

检测时间：	位置环，在读取模拟轴的反馈时。
引起原因：	由于 CAN 总线的错误，某个反馈读数、反馈模块或 CAN 总线饱和故障而不能完成模拟轴的反馈读数。
解决方案：	解决 CAN 总线或反馈模块可能出现的问题，如果总线饱和，则增加 LOOPTIME。

3712 ' 龙门轴原点搜索错误 '

检测时间：	执行程序时。搜索龙门轴原点时。
引起原因：	在从轴之前检测到主轴的原点开关。
解决方案：	调整龙门轴的原点开关位置。按下的第一个原点开关必须为从轴的。

3800 ' 连续微动进给率等于 0'

检测时间：	执行程序或切换至连续微动模式时。
引起原因：	连续微动的进给率为 0。未编写手动进给率 V.G.FMAN 且机床参数 JOGFEED 为 0。
解决方案：	检查轴机床参数 JOGFEED。

3801 ' 增量微动的距离或进给率为 0'

检测时间：	执行程序时。
引起原因：	增量微动的进给率或距离为 0。未编写手动进给率 V.G.FMAN 且机床参数 INCJOGFEED 为 0 或机床参数 INCJOGDIST 为 0。
解决方案：	检查机床参数 INCJOGFEED 和 INCJOGDIST。

3803 ' 手轮分辨率不能为 0'

检测时间：	改变手轮分辨率时。
引起原因：	通过手动面板或 PLC 为参数 MPGRESOL[i] 选择的索引为 0。
解决方案：	检查轴机床参数 MPGRESOL。

3804 ' 手轮索引超出规定范围 (转动开关位置 1-3)'

检测时间：	改变手轮分辨率时。
引起原因：	通过 PLC 设置的手轮位置超出许可范围。
解决方案：	检查从 PLC 写入 PLC.MPGIDX 的值。

3805 ' 增量微动的进给率或距离为 0'

检测时间：	改变增量微动的距离或进给率时。
引起原因：	通过 PLC 或手动面板改变了增量微动的索引。参数 INCJOGDIST 或 INCJOGFEED 选择的索引为 0。
解决方案：	检查轴机床参数 INCJOGDIST 和 INCJOGFEED。

3806 ' 增量微动进给率大于最大值 '

检测时间：	执行程序时。
引起原因：	机床参数设置的增量微动进给率大于最大许可值。
解决方案：	检查机床参数 MAXMANFEED、 INCJOGFEED 和 G00FEED。

3807 ' 增量微动索引超出规定范围 (转动开关位置 1-5)'

检测时间：	执行程序时。
引起原因：	通过 PLC 设置的增量微动位置超出许可范围。
解决方案：	检查从 PLC 写入 PLC.INCJOGIDX 的值。

3808 ' 轴不存在或不可用 '

检测时间： 执行程序时。

引起原因： 可能的原因包括：

1. 退出微动模式时，该轴并不处于微动模式中。
2. 龙门对的从轴编写了 G101 功能。
3. 龙门对的从轴编写了 G102 功能。

解决方案： 第一种情形可通过重启解决。



3809 'G95 中的主轴转速编程值为 0'

检测时间：　执行程序时。

引起原因：　G95 有效时以连续或增量微动模式移动轴，且同步主轴的编程转速为 0。

解决方案：　为同步主轴编写转速，可通过 SYNC 寄存器中的 PLC 或主轴选择。

4000-4999 号错误

4000 'SERCOS 环初始化出错 '

引起原因：	由于光纤连接问题、CNC 与驱动器参数设置错误以及 SERCOS 或驱动器面板上的问题等导致 SERCOS 环初始化过程失败。

种类

0	芯片初始化出错。 SERCOS 板未检测，或者由于循环通道中的轴与待转换的数据过多而导致 SERCON 的 DPRAM 溢出。
1	第 4 相位初始化环出错。
2	错误复位出错。

转换

以下给出初始化顺序中的出错点：

0	时间到。
1	改变至相位 0 时出错 (光纤)
2	改变至相位1时出错。(驱动器不反应: 硬件故障或选择器中ID错误)。
3	改变至相位 2 时出错。
4	读 OEM 版本出错。(CNC 的驱动器 ID 错误)。
5	读 T1min 出错。
6	读 Tatmt 出错。
7	读 T4min 出错。
8	读 Tmtsy 出错。
9	读 Tmtsc 出错。
10	读 SlaveNr 出错。
11	读 Tatat。
12	计算时间出错。
13	写 OEM 口令出错。
14	读 T1 出错。
15	读 T4 出错。
16	读 T3 出错。
17	读 T2 出错。
18	读 Tncyc 出错。
19	读 Tscyc 出错。
20	读 MDTlen 出错。
21	读 TelegramType 出错。
22	读 MDT 清单出错。
23	读 AT 清单出错。
24	读 MDT 偏置出错。
25	读 RealTime Bit 2 出错。
26	读 OpMode 出错。
27	重启指令出错。
28	停指令出错。
29	相位 3 指令出错。
30	改变至相位 3 时出错。
31	相位 4 指令出错。



32	改变至相位 4 时出错。
33	读分类诊断 1 出错。
34	缺省错误。
35	读 Tncyc 出错。
36	读 OpMode 出错。
37	读 AxisType 出错。
38	读 G00Feed 出错。
39	读 Monit Window（监视窗口）出错。
40	读 SP100 出错。
41	读 KV 出错。
42	读 Checksum 出错。
43	读 DV33 出错。
44	写 DV33 出错。

错误值 1

接收 PDU 代码出错 (必须为 5)。

错误值 2

SERCOS 驱动器错误代码：

0	SERC_NO_ERROR（无错）
0x8001	ERROR_HSTIMEOUT（错误 _HS 时间到）
0x8002	ERROR_SCHSTIMEOUT （错误 _SCHS 时间到）
0x8004	SERC_ERROR_TIMEOUT(SERC_ 错误 _ 时间到）
0x4000	ERROR_CALCULATE_T1（错误 _ 计算 _T1)
0x4001	ERROR_CALCULATE_T2（错误 _ 计算 _T2)
0x4002	ERROR_CALCULATE_T3（错误 _ 计算 _T3)
0x4004	ERROR_CALCULATE_T4（错误 _ 计算 _T4)
0x4008	ERROR_CALCULATE_TEND（错误 _ 计算 _ 趋势）
0x2001	ERROR_WRONGPHASE（错误 _ 相位错误）
0x2002	ERROR_WRONGADDRESS（错误 _ 地址错误）
0x2004	ERROR_WRONGATNUMBER（错误 _ 编号错误）
0x2008	ERROR_SCTRANSNOTREADY （错误 _SCTRANS 未准备好）
0x1002	ERROR_DPRAMOVERFLOW（错误 _DPRAM 溢出）
0x1004	ERROR_SCNOTINIT（错误 _SC 未在里面）
0x1008	ERROR_WRONGCHANNELNUMBER（错误 _ 通道号错误）
0x0801	ERROR_SCTRANSNODATA（错误 _SCTRANS 无数据）
0x0802	ERROR_SCNODATA（错误 _SC 无数据）
0x0803	NOT_READY_FOR_SCDATA(SC 数据未准备好）
0x0400	ERROR_SCTRANS （错误 _SCTRANS)
0x0201	ERROR_ATMISS（错误 _AT 缺失）
0x0202	ERROR_NERR（错误 _NERR)
0x0203	ERROR_MSTMISS （错误 _MST 缺失）
0x0204	ERROR_DISTORSION（错误 _ 失真）
0x0205	ERROR_FIBRA_ROTA（错误 _ 光纤 _ROTA)
0x0101	NOT_READY （未准备好）
0x0102	BUSYTIMEOUT（总线终止连接）
0x0080	ERROR_DEFAULT（错误 _ 缺省）
0x0040	READY_FOR_SCDATA(SC 数据准备好）

错误值 2+3

其他 SERCOS 驱动器错误代码。

5+0x10	请求退出 / 中断 / 继续执行处于非活动状态的指令。
7+0x7	请求被占用的通道。
7+0x20	逻辑轴号错误。
8+0xFE0	请求被占用的通道。

解决方案： 检查以下几项：
- 分析错误代码，确定错误源。
- 确保光纤连接正确，且状态良好。
- 检查 CNC 与驱动器的参数设置。驱动器选择器的拨动手轮位于驱动器 (DriveID)、参数 LOOPTIME、SERCPOWSEL、SERCBRATE、OPMODE 等位置。

与我公司的技术支持部门联系。

4001 'CNC 与驱动器的 LOOPTIME 参数不一致 '

引起原因： CNC 与驱动器中的 LOOPTIME 参数值不一样。

解决方案： 在 CNC 与驱动器中设置相同的参数值。

4002 'CNC 与驱动器的 OPMODE 参数不一样 '

引起原因： CNC 与驱动器中的 OPMODE 参数值不一样。

解决方案： 在 CNC 与驱动器中设置相同的参数值。

4003 'CNC 与驱动器的 AXISMODE 参数不一样 '

引起原因： CNC 与驱动器中的 AXISTYPE 参数值不一样。

解决方案： 在 CNC 与驱动器中设置相同的参数值。

4005 ' 驱动器中未激活跟随误差监视 '

引起原因： 驱动器中未激活跟随误差监视。

解决方案： 在驱动器中激活参数 PP159。

4006 ' 驱动器参数 SP100 必须为 0'

引起原因： 驱动器没有激活的附加指令。

解决方案： 设置驱动器参数 SP100 为 0。

4007 'SERCOS 主轴需要非 0 增益 '

引起原因： 主轴的 KV 等于 0。

解决方案： 为驱动器的 KV 设置一个非 0 值。

4008 ' 驱动器重启导致 SERCOS 环出错 '

引起原因： 警告：驱动器被重启。

解决方案： 忽略通过重启按钮生效的驱动器自动重启、 WinDDS (记录版本、软件重启) 或驱动器的硬件问题。

与我公司的技术支持部门联系。

4200 'SERCOS 循环通道出错 '

引起原因： 读或写循环通道出错。

解决方案： 与我公司的技术支持部门联系。

4201 'SERCOS 服务通道出错 '

引起原因： SERCOS 服务通道从以下循环启动时出错：
- 读变量：错误清单。
- 写变量：FF、 ACF、设置参数、 KV。
- 执行指令：Park (停)、改变 SET (设置)。

解决方案： 分析错误代码，确定错误源。

与我公司的技术支持部门联系。



4202 **' 驱动器使能 (DRENA) 缺失 '**

引起原因： 移动轴时， PLC 的 DRENA 信号 (使能) 失效。

解决方案： 分析 PLC 机动，确定促使 DRENA 信号失效的原因。

4203 **' 速度使能 (SPENA) 缺失 '**

引起原因： 移动轴时， PLC 的 SPENA 信号 (使能) 失效。

解决方案： 分析 PLC 机动，确定促使 SPENA 信号失效的原因。

4204 **' 重启 SERCOS 驱动器出错 '**

引起原因： 执行重启驱动器错误的指令时出错 (ID 99)。如果在驱动器报告错误时处理 CNC 重启，将从循环中执行该指令。

解决方案： 分析错误代码，确定错误源。
与我公司的技术支持部门联系。

4205 **'SERCOS 环出错 '**

值

用以确定错误源的错误代码：

0x00000002 光纤破损。

0x00000100 AT 丢失。

0x00000200 MST 丢失。

0x00008000 AT 传输错误。(光纤或驱动器重启)

0xFFFF0000 访问 SERCON 的 RAM 指令失败。

引起原因： SERCOS 总线出错，导致相位 4 丢失。

解决方案： 分析错误代码，确定错误源。
与我公司的技术支持部门联系。

4206 **'SERCOS 驱动器出错 '**

引起原因： 驱动器报告错误。

解决方案： 分析错误代码。参考驱动器手册。



5000-5999 号错误

5000 'PLC 出错：定时器不存在 '

检测时间： 读取 PLC 定时器数据过程中。
引起原因： 请求读取不存在的定时器。
解决方案： 有效定时器：T1 - T256。

5001 'PLC 出错：计数器不存在 '

检测时间： 读取 PLC 计数器数据过程中。
引起原因： 请求读取不存在的计数器。
解决方案： 有效计数器：C1 - C256。

5002 'CNCRD: 不明变量 '

检测时间： 执行 CNCRD 指令时。
引起原因： 变量不存在。
解决方案： 检查该变量语法。

5003 'CNCWR: 不明变量 '

检测时间： 执行 CNCWR 指令时。
引起原因： 变量不存在。
解决方案： 检查该变量语法。

5004 'PLC 没有该变量的读操作权限 '

检测时间： 执行 CNCRD 指令时。
引起原因： PLC 没有变量的 CNCRD 的读操作权限。
解决方案： 参考手册，确认变量的操作权限。

5005 'PLC 没有该变量的写操作权限 '

检测时间： 执行 CNCRD 指令时。
引起原因： PLC 没有变量的 CNCRD 的写操作权限。
解决方案： 参考手册，确认变量的操作权限。

5006 'PLC 写入变量的值超出规定范围 '

检测时间： 执行 CNCWR 指令时。
引起原因： 写入的值超出规定范围。
解决方案： 检查该变量的语法。

5007 'PLC 写变量时出现语法错误 '

检测时间： 执行 CNCWR 指令时。
引起原因： 变量不存在或没有该变量的写操作权限。
解决方案： 检查该变量的语法。

5008 'PLC 不能写该变量 '

检测时间： CNCWR 指令中有 GUP/LUP/LUPACT 值。
引起原因： 请求写不可用的 GUP/LUP/LUPACT 值。
解决方案： 检查全局与局部参数的设置以及 PLC 程序。

5009 'PLC 读变量时出现语法错误 '

检测时间： 执行 CNCRD 指令时。

引起原因： 变量不存在或没有该变量的读操作权限。

解决方案： 检查该变量的语法。

5010 'PLC 中除数为 0'

检测时间： 在 PLC 程序中执行 DVS/MDS 指令时。

引起原因： DVS/MDS 操作的分母为 0。

解决方案： 检查 PLC 程序，确保分母不为 0。

5013 ' 读 PLC 数字输入出错 '

检测时间： 读 PLC 的数字输入时。

引起原因： 数字 I/O 配置表错误。

解决方案： 检查数字 I/O 配置表。
与我公司的技术支持部门联系。

5014 ' 写 PLC 数字输出出错 '

检测时间： 写 PLC 的数字输出时。

引起原因： 在未完成前一数字输出的写操作前，请求写数字输出，或者数字 I/O 配置表无效。

解决方案： 进行相关检查，确保 BUS CAN 的完整性。增加 PLCFREQ 循环时间。检查数字 I/O 配置表。
与我公司的技术支持部门联系。

5015 ' 不能从 PLC 读取变量 '

检测时间： CNCRD 指令中有 GUP/LUP/LUPACT 值 。

引起原因： 请求读不可用的 GUP/LUP/LUPACT 值。

解决方案： 检查全局与局部参数的设置以及 PLC 程序。

5016 ' 读取的值超出规定范围 '

检测时间： CNCRD 指令。

引起原因： 读取的数据超出规定范围。

解决方案： 检查指令的语法。

5017 'CNCEX 程序段为空 '

检测时间： CNCEX 指令。

引起原因： 程序段为空，或寻址的通道不可用。

解决方案： 检查程序以及通道状态。

5018 '' 通信标志设置为 "1"，导致 CNCEX 未执行 '

检测时间： 执行 CNCEX 指令时。

引起原因： 开始执行新的 CNCEX 程序段时， CNCEX 程序段的控制标志设为 1 。可能是因为编程错误或通道正在处理前一 CNCEX 指令。

解决方案： 在程序中检查 CNCEX 的逻辑以及控制标志。

5020 'CNCEX: 执行不完全 '

检测时间： 执行 CNCEX 指令时。

引起原因： 不能在请求的通道中执行程序段。

解决方案： 检查执行 CNCEX 指令的通道的状态。

5021 '#'CNCEX: 指定的通道并非 PLC 通道 '

检测时间： 执行 CNCEX 指令时。

引起原因： CNCEX 程序段寻址的通道并非 PLC 通道。

解决方案： 检查决定通道类型的机床参数以及 PLC 程序。

6000-6999 号错误

6000 ' 反馈报警 '

检测时间： 执行程序时。

引起原因： 模拟轴反馈出错。它通过机床参数 FBACKAL 激活。

差分 TTL：

- 一根或多根电缆破损 (A、 B 信号或它们的反相信号)。
- 反馈输入未与计数器连接。

差分正弦信号：

- 一根或多根电缆破损 (A、 B 信号或它们的反相信号)。
- 反馈输入未与计数器连接。
- 输入信号振幅高于 1.45 Vpp (约值)
- 输入信号振幅低于 0.4 Vpp (约值)
- A 与 B 信号相移过大 (正交，理论上)

对于非差分信号，必须禁止反馈报警 (FBACKAL)，因为不能很确切的进行判断。

解决方案： 与我公司的技术支持部门联系。

6001 ' 趋势检验报警被激活 '

检测时间： 轴失控且趋势检验检测被激活时。

引起原因： 轴失控。轴正反馈的时间大于参数 ESTDELAY 设定的值。

解决方案： 检查速度指令与反馈的标记。设置参数 ESTDELAY、指令标记 AXISCHG 以及反馈 LOOPCHG。

6002 'PLC 程序未运行 '

检测时间： 连续监控 PLC 状态时。

引起原因： PLC 程序未运行。

- 安装新版软件。
- PLC 停止且未重新开始。

解决方案： 设置 PLC 程序运行。若有必要，编译 PLC 程序。

6003 ' 外部紧急事件被激活 '

检测时间： 连续监控每个插补循环时。

引起原因： PLC 标志 _EMERGEN 被设为 0。

解决方案： 检查 PLC 程序中 _EMERGEN 信号的逻辑。

6004 ' 初始化 BUS CAN 时出错 '

引起原因： 驱动器报告错误。

解决方案： 分析错误代码。参考驱动器手册。

6005 ' 选择 BUS CAN 工作频率时出错 '

检测时间： 系统启动时。

引起原因： 一个或几个节点未调谐至参数设定的频率。

解决方案： 检查下述几项：

- CAN 电缆距离与频率参数匹配。
- 是否所有的模块在诊断模式中都能被识别。

与我公司的技术支持部门联系。



6006 'CAN 节点不响应 '

引起原因：由于重启、短路、性能较差等原因，使得一个或几个远程节点不响应。

解决方案：是否所有的模块在诊断模式中都能被识别。
与我公司的技术支持部门联系。

6007 'COMPCI 中的 Watchdog（看门狗）'

引起原因：由于较差的性能、触点等，使得 COMPCI 不响应。

解决方案：与我公司的技术支持部门联系。

6008 'PC + COMPCI 中的 Watchdog（看门狗）'

引起原因：由于较差的性能、触点等，使得 COMPCI 和 PC 不响应。

解决方案：与我公司的技术支持部门联系。

6009 'PC 中的 Watchdog（看门狗）'

引起原因：PC 不响应。

解决方案：与我公司的技术支持部门联系。

6010 'COMPCI 处理出错 '

引起原因：由 COMPCI 管理的步骤中出现各种错误。

解决方案：与我公司的技术支持部门联系。

6011 ' 远程节点中出现 CAN 错误 '

引起原因：节点报告错误。

错误代码

1	CAN 控制器溢出。接收的信息可能丢失。
2	BusOFF（总线关闭）。节点检测到总线 (BUS) 掉线。
3	接收先入先出溢出。接收的信息可能丢失。
4	警告。错误计数器超出 1 级 (96)。
11	节点报告 COMPCI 不响应当前的控制器。
12	节点重启 (电源问题、短路、看门狗等。)
13	传输信息失败。

模块类型

1	模拟输出。
2	计数器。
3	数字输出。
4	数字输入。
5	模拟输入。
7	PT100 输入。
8	CAN 键盘手轮。
9	探头。
10	微动键盘。
11	字母数字键盘。

节点逻辑号

解决方案：与我公司的技术支持部门联系。

6012 'CAN 控制器错误 '

引起原因：COMPCI 检测到 BUSOFF（总线关）。总线掉线。

解决方案：在诊断模式中核实所有模块以及 CAN 错误计数器可被辨识。万一出错，进行相关检查，确保 BUS CAN 的完整性。
与我公司的技术支持部门联系。

6013 ' 初始化 CAN 时间到 '

检测时间： 启动时。

引起原因： 由于总线问题导致 BUS CAN 初始化失败。关 / 开顺序过快。

解决方案： 完成以下检查并采取下述措施，以确保 BUS CAN 的完整性：

- 没有与拨动手轮所选择的地址相同的 CAN 组。
- COMPCI 的 CAN 拨动手轮必须为 0。
- 行终止符。
- 接地。
- CAN 电缆连续性。
- CAN 电缆连接器 (拔下插头然后在插上电源)。
- 节点与电源之间的带状 (扁平) 电缆连接。
- CAN 组的电源 (电平、可能的重启等)。
- 是否所有的模块在诊断模式中都能被识别。

与我公司的技术支持部门联系。

6014 ' 访问 COMPCI 的 DPRAM 失败 '

检测时间： 启动时。

引起原因： 由于总线问题导致 BUS CAN 初始化失败。关 / 开顺序过快。

解决方案： 完成以下检查并采取下述措施，以确保 BUS CAN 的完整性：

- 没有与拨动手轮所选择的地址相同的 CAN 组。
- COMPCI 的 CAN 拨动手轮必须为 0。
- 行终止符。
- 接地。
- CAN 电缆连续性。
- CAN 电缆连接器 (拔下插头然后在插上电源)。
- 节点与电源之间的带状 (扁平) 电缆连接。
- CAN 组的电源 (电平、可能的重启等)。
- 是否所有的模块在诊断模式中都能被识别。

与我公司的技术支持部门联系。

6015 'CAN 错误计数器超出 1 级 '

引起原因： (Rx/Tx) 错误计数器超出 1 级 (96)。

解决方案： 进行相关检查，确保 BUS CAN 的完整性。
与我公司的技术支持部门联系。

6016 'CAN 错误计数器处于临界值 '

引起原因： T(Rx/Tx) 错误计数器超出 1 临界值 (127)。

解决方案： 进行相关检查，确保 BUS CAN 的完整性。
与我公司的技术支持部门联系。

6017 'CAN 控制器先入先出溢出 '

引起原因： CAN 控制器接收 FIFO（先入先出）产生溢出。可能会丢失接收到的信息。

解决方案： 与我公司的技术支持部门联系。

6018 'COMPCI 的 FIFO（先入先出）CAN 溢出 '

引起原因： COMPCI 的接收 FIFO（先入先出）产生溢出。可能会丢失接收到的信息。

解决方案： 与我公司的技术支持部门联系。

6019 'CAN 读操作时间到 '。

引起原因： 读取数字与模拟输入、计数器以及键盘手轮失败。节点未及时送出信息。

解决方案： 在诊断模式中核实所有模块以及 CAN 错误计数器可被辨识。万一出错，进行相关检查，确保 BUS CAN 的完整性。若有必要，增加循环时间 (LOOPTIME、PLCFREQ)。
与我公司的技术支持部门联系。

6020 'CAN 循环重叠 '

引起原因： 读取数字与模拟输入、计数器以及键盘手轮失败。节点未及时送出信息。

解决方案： 在诊断模式中核实所有模块以及 CAN 错误计数器可被辨识。万一出错，进行相关检查，确保 BUS CAN 的完整性。若有必要，增加循环时间 (LOOPTIME、PLCFREQ)。
与我公司的技术支持部门联系。



6021 'CAN 传输存在问题 '

引起原因： 传输数字与模拟输出等失败。可能是 BUS(总线) 崩溃。

解决方案： 在诊断模式中核实所有模块以及 CAN 错误计数器可被辨识。万一出错，进行相关检查，确保 BUS CAN 的完整性。若有必要，增加循环时间 (LOOPTIME、PLCFREQ)。
与我公司的技术支持部门联系。

6022 ' 初始化轴计数器出错 '

检测时间： 启动时。

引起原因： 计数器不存在。

解决方案： 检查参数。
与我公司的技术支持部门联系。

6023 ' 初始化手轮计数器出错 '

检测时间： 启动时。

引起原因： 计数器不存在。

解决方案： 检查参数。
与我公司的技术支持部门联系。

6024 ' 初始化键盘手轮时出错 '

检测时间： 启动时。

引起原因： 手轮输入不存在。

解决方案： 检查参数。
与我公司的技术支持部门联系。

6025 'COMPCI 校验和错误 '

引起原因： 通过指令 RAM 将数字或模拟输出数据从 PC 机发送至 COMPCI，其校验和控制器出错。

模块类型

1	模拟输出。
3	数字输出。

解决方案： 与我公司的技术支持部门联系。

6026 'COMPCI 硬件错误 '

检测时间： 系统启动时。

引起原因： 进行 COMPCI 硬件测试时出错。

解决方案： 诊断模式显示该类错误的详细信息。确保 COMPCI 的软件版本正确。
与我公司的技术支持部门联系。

6027 ' 远程 CAN 节点存在硬件错误 '

检测时间： 系统启动时。

引起原因： 进行远程节点硬件测试时出错。

解决方案： 诊断模式显示该类错误的详细信息。确保节点的软件版本正确。
与我公司的技术支持部门联系。

6028 ' 访问 SERCON 的 RAM 出错 '

检测时间： 启动时。

引起原因： RAM 测试故障。

解决方案： 与我公司的技术支持部门联系。

6029 'CNC 的位置增量超出极限 '

引起原因： 模拟轴的位置增量过大。计数器节点硬件故障，访问 COMPCI (线路连接) 等出错。

解决方案： 与我公司的技术支持部门联系。

6030 'CAN 计数器节点的位置增量超出极限 '

引起原因： 模拟轴的位置增量过大。计数器节点硬件故障。

解决方案： 与我公司的技术支持部门联系。



6031 'COMPCI/ 远程节点软件版本不相容 '

检测时间： 启动时。

引起原因： CNC 与 COMPCI 或远程模块的软件版本不相容。

解决方案： 更新 COMPCI 和远程模块的软件版本 (itfcboot.exe)。





7000-7999 号错误

7001 ' 循环刀库中有两把连续刀具 T(需要 M6)'

检测时间： 执行程序时。
引起原因： 编写了循环刀库与两把连续刀具 T。
解决方案： 在每个 T 后编写一条 M6 功能。

7002 ' 刀具不在刀库中且不允许使用基础刀具 '

检测时间： 执行程序时。
引起原因： 程序中编写了刀具表中的某把刀，但是刀库表中没有这把刀。这种情况下，该刀被视作基础刀具，但是机床参数显示不允许使用基础刀具。
解决方案： 改变机床参数，认可基础刀具。

7003 ' 表中未定义 T'

检测时间： 执行程序时。
引起原因： 编写了刀具表中未定义的刀具。
解决方案： 定义刀具。

7004 ' 刀具被拒绝或破损 (超过使用寿命)'

检测时间： 执行程序时。
引起原因： 编写了基础刀具，但是该刀超过使用寿命 (破损) 或无效。
解决方案： 为超过使用寿命或被拒绝的刀具编写等效的刀具来进行替换。

7005 ' 刀具被拒绝或超过使用寿命，且无替换刀具 '

检测时间： 执行程序时。
引起原因： 编写了破损 (超过使用寿命) 或无效的刀具。搜索不到同系列刀具。
解决方案： 使能同系列的刀具，这样便可以使用该刀。

7006 ' 该刀具不允许使用 D'

检测时间： 执行程序时。
引起原因： 编写了无效刀刃。刀具的刀刃小于编程值。
解决方案： 为刀具定义必要的刀刃。

7007 ' 加载模式中：不允许使用 T0'

检测时间： 执行程序时。
引起原因： 刀库处于加载模式时，不能编写 T0。
解决方案： 编写将加载的刀具 T。

7008 ' 加载模式中：该刀具已经被加载 '

检测时间： 执行程序时。
引起原因： 加载模式中，编程刀具已经位于刀库中。
解决方案： 编写未被加载的刀具。

7009 ' 加载模式中：M6 功能所在程序段中没有 T'

检测时间： 执行程序时。
引起原因： 加载模式中，编写的 M6 功能中没有对应的刀具 T。
解决方案： 首先为待加载的刀具编写 T。



7010 ' 加载模式中：T 对应两个 M6'

检测时间： 执行程序时。
引起原因： 加载模式中，同一刀具 T 检测到两个 M6 功能。
解决方案： 依次编写 T 与 M6 功能。

7011 ' 加载模式中：D 不能位于单独程序段中 '

检测时间： 执行程序时。
引起原因： 加载模式中， "D" 不能位于单独程序段中。
解决方案： 不能这样编程。

7012 ' 加载模式中：刀具破损 (超过使用寿命) 或被拒绝 '

检测时间： 执行程序时。
引起原因： 加载模式中，待加载的刀具破损 (超过使用寿命) 或被拒绝。
解决方案： 编写另一把可用刀具。

7013 ' 加载模式中：位置错误或刀库中没有空间 '

检测时间： 执行程序时。
引起原因： 加载模式中，检测到刀库已满或指定位置被占用。
解决方案： 如果刀库中没有空间，先卸载一把刀具。如果指定位置被占用，选择另一位置。

7014 ' 卸载模式中：T 不在刀库中 '

检测时间： 执行程序时。
引起原因： 刀库处于卸载模式时，编程刀具不在刀库中。
解决方案： 不能卸载未被加载的刀具。

7015 ' 卸载模式中：D 不能与 T 一起使用 '

检测时间： 执行程序时。
引起原因： 卸载模式中不能编写 D。
解决方案： 从编程指令中移除 D。

7016 ' 卸载模式中：D 不能位于单独行中 '

检测时间： 执行程序时。
引起原因： 卸载模式中， "D" 不能位于单独程序段中。
解决方案： 不能这样编程。

7017 ' 卸载模式中：M6 功能所在程序段中没有 T'

检测时间： 执行程序时。
引起原因： 卸载模式中，编写的 M6 功能中没有对应的刀具 T。
解决方案： 首先为待卸载的刀具编写 T。

7018 ' 卸载模式中：T 对应两个 M6'

检测时间： 执行程序时。
引起原因： 卸载模式中，同一刀具 T 检测到两个 M6 功能。
解决方案： 依次编写 T 与 M6 功能。

7019 ' 设置模式中：不允许使用 T0'

检测时间： 执行程序时。
引起原因： 设置模式中，刀具 T0 无效。
解决方案： 编写非 0 号刀具。

7020 ' 设置模式中：需要 D'

检测时间： 执行程序时。
引起原因： 设置模式中，必须指定刀具刀刃。
解决方案： 编程刀刃。



7021 ' 设置模式中：D 不能位于单独行中 '

检测时间： 执行程序时。
引起原因： 卸载模式中， "D" 不能位于单独程序段中。
解决方案： 不能这样编程。

7022 ' 设置模式中：M6 未确认 '

检测时间： 执行程序时。
引起原因： 设置模式中，未编写 M6 功能。
解决方案： 不能这样编程。

7023 ' 加载模式中：需要 M6'

检测时间： 执行程序时。
引起原因： 加载模式中编写了两把连续刀具 T。
解决方案： 在该模式中，必须在刀具 T 后编写一条 M6 功能。

7024 ' 卸载模式中：T0 未确认 '

检测时间： 执行程序时。
引起原因： 刀库处于卸载模式时，不能编写 T0。
解决方案： 编写待卸载刀具 T。

7025 ' 卸载模式中：需要 M6'

检测时间： 执行程序时。
引起原因： 卸载模式中编写了两把连续刀具 T。
解决方案： 在该模式中，必须在刀具 T 后编写一条 M6 功能。

7026 ' 卸载模式中：该刀具已被加载 '

检测时间： 执行程序时。
引起原因： 卸载模式中，编程刀具不在刀库中。
解决方案： 编写已经位于刀库中的刀具 T。

7027 'M6 功能所在程序段中没有 T'

检测时间： 执行程序时。
引起原因： 检测到 M6 功能没有对应的刀具 T。这种情形下，可通过机床参数的值来确定下一步可采取的措施 。If = 0，无错； if = 1，发出警告； if = 2，出错。
解决方案： 首先编写刀具 T 或改变参数值。

7028 'T0 未确认 '

检测时间： 执行程序时。
引起原因： 转塔刀库中不允许使用 T0。该类刀库通过旋转转塔来实现换刀。只需编写刀具 T。
解决方案： 不要编写刀具 T0。

7029 ' 不能执行操作：刀具管理器错误 '

检测时间： 执行程序时。
引起原因： 刀库紧急事件。可能是由于 PLC 激活了无效的机动标志，也有可能是 PLC 激活了 SETTMEM 或存在逻辑内部错误。
解决方案： 换刀时由于无效机动而出错。这种情形下的解决方案，是在 PLC 程序中纠正换刀机动。如果是因为 SETTMEM 出错，检查 PLC 为什么设置紧急信号。

7030 'M6 功能所在程序段中没有 T'

检测时间： 执行程序时。
引起原因： 检测到 M6 功能没有对应的刀具 T。如果参数 "M6 ALONE = NO"，则出错。
解决方案： 不能编写无对应刀具 T 的 M6 功能。

7031 'T 对应两个 M6'

检测时间： 执行程序时。
引起原因： 同一刀具 T 检测到两个 M6 功能。
解决方案： 核对程序。



7032 'PLC 操作错误 '

检测时间： 执行程序时。
引起原因： PLC 程序中的顺序错误。
解决方案： 检查 PLC 程序。

7033 'PLC 产生刀库紧急事件 '

检测时间： PLC 运行时。
引起原因： PLC 为刀库设置紧急事件标志。
解决方案： 删除紧急事件。

7034 ' 操作错误：刀库中没有空间 '

检测时间： 执行程序时。
引起原因： PLC 机动生成错误。
解决方案： 检查 PLC 程序。

7035 ' 刀库表错误 '

检测时间： 执行程序时。
引起原因： 机动过程中刀库出错。
解决方案： 检查刀库状态。也许是刀库中没有空间。

7036 ' 不能改变刀库或主轴上刀具的系列 '

检测时间： 编辑表格或执行程序时。
引起原因： 出于安全考虑，不能改变刀库中刀具的系列。
解决方案： 首先卸载该刀具。

7037 ' 加载与卸载模式下，主轴必须为空 '

检测时间： 执行程序时。
引起原因： 改变刀库模式至加载或卸载模式，主轴不为空 (或者刀库夹臂不为空)。
解决方案： 使主轴以及夹臂为空。

7039 ' 刀具不能同时位于主轴和夹臂 2 上 '

检测时间： 执行程序时。
引起原因： 使用同步或异步刀库，且其所带的换刀器同时需要位于夹臂 2 上的相同刀具，才会出现该特定情形。
解决方案： 该情形不常出现，一般会在某个错误后出现。可编写 T0 M6 使刀具返回刀库来纠正该问题，这样便可恢复正常。

7040 ' 操作错误：先空出夹臂 1'

检测时间： 执行程序时。
引起原因： 使用同步或异步刀库，且其所带的换刀器同时需要位于夹臂 1 上的相同刀具，才会出现该特定情形。
解决方案： 该情形不常出现，一般会在某个错误后出现。可编写 T0 M6 使刀具返回刀库来纠正该问题，这样便可恢复正常。

7041 ' 无刀具处于活动状态 '

检测时间： 执行程序时。
引起原因： 编写了刀刃，但是主轴上没有活动刀具。
解决方案： 在主轴上安置刀具。

7042 ' 编写了 POS，但是刀库不在加载模式中 '

检测时间： 执行程序时。
引起原因： 编写了刀具加载位置，但是刀库并不处于加载模式。
解决方案： 只在加载模式中才允许编程加载位置。

7043 ' 请求的刀具为另一通道中的活动刀具 '

检测时间： 执行程序时。
引起原因： 编写的刀具已经为另一通道中的活动刀具。
解决方案： 从另一通道的主轴上卸载请求的刀具，可在通道中编写 T0 M6 实现。

8000-8999 号错误

8200 ' 辨识整数时词法出错 '

检测时间： 在 MDI 模式中编辑和执行程序时。

引起原因： 将字符串转换为数字值时出错。通常当指令或变量中编写的某些值错误时才会出现该错误。

解决方案： 查阅编程手册，确认指令中的参数许可值以及变量的索引。

8201 ' 辨识浮点数时词法出错 '

检测时间： 在 MDI 模式中编辑和执行程序时。

引起原因： 将字符串转换为数字值时出错。通常当指令或变量中编写的某些值错误时才会出现该错误。

解决方案： 查阅编程手册，确认指令中的参数许可值以及变量的索引。

8203 ' 缺失注释开始字符 '

检测时间： 编辑程序时。

引起原因： 检测到注释结束字符，但是前面没有注释开始字符。

解决方案： 确定注释同时具有开始与结束字符 "(" 与 ")"。

8204 ' 缺失 $ 或 #'

检测时间： 在 MDI 模式中编辑和执行程序时。

引起原因： 可能的原因包括：
- 编程指令没有 "#"。
- 编程的流程控制指令没有 "$"。

解决方案： 所有指令必须以 "#" 字符开始，所有流程控制指令必须以 "$" 字符开始。

8205 ' 缺失 $ '

检测时间： 编辑程序时。

引起原因： 编写的流程控制指令没有前置字符 "$"。

解决方案： 在控制指令名前编写 "$" 字符。

8206 ' 缺失 # '

检测时间： 编辑程序时。

引起原因： 编程指令前没有前置字符 "#"。

解决方案： 在指令名前编写 "#" 字符。

8207 ' 程序或子程序名过长 '

检测时间： 编辑程序时。

引起原因： 程序或子程序名的字符数超出最大许可值。

解决方案： 允许使用的最大字符数为 14。

8209 ' 活动轴错误 '

检测时间： 编辑程序时。

引起原因： 轴名中有错误字符。

解决方案： 带万用字符的轴名为 @1 ～ @5。



8210 ' 超出整数极限 '

检测时间：编辑程序时。
引起原因：编程整数值过大。
解决方案：最大整数值为 4294967295。

8211 ' 超出整数部分的极限 '

检测时间：编辑程序时。
引起原因：数字的整数部分的值错误。
解决方案：整数部分的有效值范围为 ±99999。

8212 ' 超出小数部分的极限 '

检测时间：编辑程序时。
引起原因：数字的小数部分的值错误。
解决方案：小数部分的有效值范围为 ±0.99999。

8213 ' 超出小数部分的格式 '

检测时间：编辑程序时。
引起原因：超出数字中小数的最大数值。
解决方案：小数中允许使用的最大数字为 5。

8214 ' 流程控制指令未知 '

检测时间：编辑程序时。
引起原因："$" 后的编程指令错误。
解决方案：检查指令语法。

8218 ' 字符错误 '

检测时间：编辑程序时。
引起原因：检测到程序段中有无效字符。
解决方案：检查程序段语法。

8221 ' 语法错误 '

检测时间：在 MDI 模式中编辑和执行程序时。
引起原因：编程指令或变量的语法错误。
解决方案：查阅编程手册，检查指令或变量的语法。

8222 'M 功能不存在 '

检测时间：编辑程序时。
引起原因：编写的 M 功能不存在。
解决方案：查阅编程手册，检查 M 功能。

8223 'M 功能互斥或相同 '

检测时间：在 MDI 模式中编辑和执行程序时。
引起原因：可能的原因包括：
- 同一 M 功能在程序段中重复编程。
- 同一程序段中编写了不相容的 M 功能。

解决方案：解决的办法有：
- M 功能只能在程序段中编写一次。
- 参考编程手册，检查 M 功能的不相容性。

8224 'M 功能超出规定范围 '

检测时间：编辑程序时。
引起原因：编写的 M 功能不存在。
解决方案：查阅编程手册，检查 M 功能。

8225 'G 功能超出规定范围 '

检测时间：在 MDI 模式中编辑和执行程序时。
引起原因：编写的 G 功能不存在。
解决方案：查阅编程手册，检查 G 功能。

8226 'H 功能超出规定范围 '

检测时间： 编辑程序时。
引起原因： 编写的 H 功能不存在。
解决方案： 查阅编程手册，检查 H 功能。

8227 ' 只有 G63 允许使用负的 S'

检测时间： 编辑程序时。
引起原因： 程序段中编写了负的主轴转速，但是未编写 G63 功能。
解决方案： 主轴转速必须为正值。只有在编写了 G63 的程序段中才允许使用负的转速。

8228 ' 刀具号超出规定范围 '

检测时间： 编辑程序时。
引起原因： 编写了负的刀具号。
解决方案： 刀具号必须为 0 或正值。

8230 ' 进给率 F 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 F 功能。
解决方案： 只在程序段中编写一次 F 功能。

8232 '"A" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 A 轴。
解决方案： 只在程序段中编写一次 A 轴。

8233 '"B" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 B 轴。
解决方案： 只在程序段中编写一次 B 轴。

8234 '"C" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 C 轴。
解决方案： 只在程序段中编写一次 C 轴。

8235 '"U" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 U 轴。
解决方案： 只在程序段中编写一次 U 轴。

8236 '"V" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 V 轴。
解决方案： 只在程序段中编写一次 V 轴。

8237 '"W" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 W 轴。
解决方案： 只在程序段中编写一次 W 轴。

8238 '"X" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 X 轴。
解决方案： 只在程序段中编写一次 X 轴。



8239 '"Y" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 Y 轴。
解决方案： 只在程序段中编写一次 Y 轴。

8240 '"Z" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 Z 轴。
解决方案： 只在程序段中编写一次 Z 轴。

8241 ' 参数超出规定范围 '

检测时间： 执行程序时。
引起原因： 指令的参数的编程值过大。
解决方案： 为指令的参数编写较小的值。

8242 ' 缺失括弧 '

检测时间： 在 MDI 模式中编辑和执行程序时。
引起原因： 可能的原因包括：
- 编程指令的参数必须位于括弧之间。
- 变量需要在括弧之间编写索引。

解决方案： 查阅编程手册，检查指令或变量的语法。

8243 '"I" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 I 轴。
解决方案： 只在程序段中编写一次 "I" 值。

8244 '"J" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 J 轴。
解决方案： 只在程序段中编写一次 "J" 值。

8245 '"K" 重复编程 '

检测时间： 编辑程序时。
引起原因： 同一程序段中重复编写 K 轴。
解决方案： 只在程序段中编写一次 "K" 值。

8247 ' 表达式中只能使用 R1'

检测时间： 编辑程序时。
引起原因： 半径值编程错误。
解决方案： 半径只能使用 "R" 或 "R1" 进行编程。

8250 ' 轴变量无写操作权限 '

检测时间： 编辑程序时。
引起原因： 试图写没有写操作权限的轴变量。
解决方案： 只能读该变量。

8251 ' 全局变量无写操作权限 '

检测时间： 编辑程序时。
引起原因： 试图写没有写操作权限的全局变量。
解决方案： 只能读该变量。

8253 'G20 中缺轴 '

检测时间： 编辑程序时。
引起原因： 没有编写完 G20 功能需要的所有参数。
解决方案： 检查 G20 的编程语法。

8254　'G20 中的轴过多 '

检测时间：　编辑程序时。

引起原因：　G20 中编写的某些参数不允许使用。

解决方案：　检查 G20 的编程语法。

8256　' 插补循环指令不存在 '

检测时间：　在 MDI 模式中编辑和执行程序时。

引起原因：　编程指令不存在。

解决方案：　参考编程手册。

8257　' 程序或子程序名后不能有其他内容 '

检测时间：　编辑程序时。

引起原因：　可能的原因包括：

1. 在定义主程序或子程序名的程序段中，还有其他字符。
2. 主程序或子程序名中有空格。

解决方案：　主程序或子程序的说明只能通过注释实现。禁用的字符包括：

1. 主程序：\ / : * ? " < > | 以及空格。
2. 局部子程序：/ ? " < > |) 以及空格。

8258　'#TIME 后需要表达式 '

检测时间：　在 MDI 模式中编辑和执行程序时。

引起原因：　#TIME 指令编程错误。

解决方案：　可使用 #TIME 指令以两种不同方法编写暂停：

- #TIME [< 表达式 >]
- #TIME [< 表达式 >]

参数 < 表达式 > 可以为整数、算术参数或控制变量。

8265　' 单词未知或不完整 '

检测时间：　编辑程序时。

引起原因：　可能的原因包括：

1. 编程功能、指令或表达式无效。
2. 特定轴编写的 M 功能错误。
3. #TOOL AX 指令编程错误。

解决方案：　查阅编程手册，检查指令、功能或表达式的语法。

8267　' 循环变量无写操作权限 '

检测时间：　编辑程序时。

引起原因：　试图写没有写操作权限的循环变量。

解决方案：　只能读该变量。

8275　' 同一程序段中的 M 功能过多 '

检测时间：　编辑程序时。

引起原因：　同一程序段中的 M 功能过多。

解决方案：　程序段中允许使用的 M 功能最多为 7 个。

8276　' 同一程序段中的 H 功能过多 '

检测时间：　编辑程序时。

引起原因：　同一程序段中的 H 功能过多。

解决方案：　程序段中允许使用的 H 功能最多为 7 个。



8279 ' 镜像重复 '

检测时间： 编辑程序时。

引起原因： 可能的原因包括：
1. 同一程序段中编写了 G11、 G12 或 G13 功能中的两个或两个以上。
2. 同一程序段中编写了 G10 和 G11、 G12 或 G13 功能。
3. 同一程序段中编写了 G14 和 G11、 G12 或 G13 功能。

解决方案： 解决的办法有：
1. G11、G12 和 G13 功能可在同一程序段中组合使用，但是任意一种功能都只能编写一次。
2. G10 功能不能编写在 G11、 G12 或 G13 所在的程序段中。
3. G14 功能不能编写在 G11、 G12 或 G13 所在的程序段中。

8280 ' 主轴定位转速为负 '

检测时间： 编辑程序时。

引起原因： M19 中编写的主轴转速为负 (Sx.POS)。

解决方案： 定位转速必须为正。

8281 ' 主轴定位转速重复编程 '

检测时间： 在 MDI 模式中编辑和执行程序时。

引起原因： M19 中的主轴定位转速在同一程序段中重复编程。

解决方案： 只编写一次主轴定位转速 "S.POS"。

8282 ' 循环参数重复 '

检测时间： 编辑程序时。

引起原因： 固定循环参数在程序段中重复编程。

解决方案： 每个固定循环参数只能在程序段中编写一次。

8283 ' 固定循环参数错误 '

检测时间： 编辑程序时。

引起原因： 固定循环中编写的参数错误。

解决方案： 查阅编程手册，确定固定循环需要和允许使用的参数。

8284 'PLC 变量无写操作权限 '

检测时间： 编辑程序时。

引起原因： 试图写没有写操作权限的 PLC 变量。

解决方案： 只能读该变量。

8285 'G20：只有纵轴允许使用负标记 '

检测时间： 编辑程序时。

引起原因： 非纵轴在 G20 功能中编写了负标记。

解决方案： G20 中，只有纵轴才能拥有负标记。纵轴为包含刀具，且在参数 3 或 5 中指定的轴。

8290 ' 刀具管理器变量无写操作权限 '

检测时间： 编辑程序时。

引起原因： 试图写没有写操作权限的刀具管理器变量。

解决方案： 只能读该变量。

8291 ' 机床参数变量无写操作权限 '

检测时间： 编辑程序时。

引起原因： 试图写没有写操作权限的机床参数变量。

解决方案： 只能读该变量。

8297 ' 参数重复 '

检测时间： 编辑程序时。

引起原因： 可能的原因包括：

1. #HSC 指令中的 CONTERROR 参数重复。
2. 在子程序调用指令 #PCALL、#MCALL 或有关联子程序的 G 功能中，某些参数重复编程。
3. #PROBE 或 #POLY 指令中的某些参数重复编程。

解决方案： 每个参数在程序段中只编写一次。

8299 ' 参数列表缺失括弧 '

检测时间： 在 MDI 模式中编辑和执行程序时。

引起原因： 编程指令的参数列表必须位于圆括号中。

解决方案： 查阅编程手册，检查指令语法。

8300 'G170、 G171 和 G157 至少需要一根轴 '

检测时间： 编辑程序时。

引起原因： G170、 G171 或 G157 功能所在程序段未编写任何轴。

解决方案： 编写受 G170、 G171 或 G157 功能影响的轴。

8302 'X 最小坐标必须小于 X 最大坐标 '

检测时间： 编辑程序时。

引起原因： #DGWZ 指令中编写的 X 最小坐标等于或大于 X 最大坐标。

解决方案： 最小坐标必须小于最大坐标。

8303 'Y 最小坐标必须小于 Y 最大坐标 '

检测时间： 编辑程序时。

引起原因： #DGWZ 指令中编写的 Y 最小坐标等于或大于 Y 最大坐标。

解决方案： 最小坐标必须小于最大坐标。

8304 'Z 最小坐标必须小于 Z 最大坐标 '

检测时间： 编辑程序时。

引起原因： #DGWZ 指令中编写的 Z 最小坐标等于或大于 Z 最大坐标。

解决方案： 最小坐标必须小于最大坐标。

8306 'G201 需要编写专用功能 #AXIS'

检测时间： 编辑程序时。

引起原因： G201 功能所在程序段中未编写 #AXIS 功能。

解决方案： G201 功能所在程序段中必须编写 #AXIS 指令。该指令必须指定受 G 功能影响的轴。

8307 ' 第三原始轴编程错误 '

检测时间： 编辑程序时。

引起原因： G20 指令中参数 5 的编程值与参数 1 或参数 2 相同。

解决方案： 参数 5 必须与参数 1 和参数 2 不同。

8308 'G04 后需要表达式或 K'

检测时间： 编辑程序时。

引起原因： G4 功能编程错误。

解决方案： G4 功能编程如下：

1. G4 < 暂停时间 >
2. G4 < 暂停时间 >

8309 ' 编程轴过多 '

检测时间： 编辑程序时。

引起原因： #POLY 指令中编写的轴过多。

解决方案： 可编程轴的最大数量为 3。



8310 ' 所需参数缺失 '

检测时间： 编辑程序时。

引起原因： 可能的原因包括：

1. #POLY 指令中未编写某些必需的参数。
2. #CS DEF、#CS ON、#CS NEW、#ACS DEF、#ACS ON 或 #ACS NEW 指令中未编写某些必需的参数。
3. #PROBE 指令中未编写某些必需的参数。

解决方案： 查阅编程手册，检查指令语法。

8311 ' 参数值错误 '

检测时间： 编辑程序时。

引起原因： 可能的原因包括：

1. #CS DEF、#CS ON、#CS NEW、#ACS DEF、#ACS ON 或 #ACS NEW 指令中的某些自变量编程值错误。
2. #HSC 指令中的 CONTERROR 编程值错误。
3. #POLY 指令中的 EP 或 R 的编程值错误。
4. 算术参数的索引编程值错误。

解决方案： 查阅编程手册，检查指令语法。算术参数的索引必须为 0 或正值。

8312 '#CS/#ACS: 模式错误 '

检测时间： 编辑程序时。

引起原因： #CS DEF、#CS ON、#CS NEW、#ACS DE、#ACS ON 或 #ACS NEW 指令中的模式（MODE）编程值错误。

解决方案： 查阅编程手册，检查指令语法。

8313 '#CS/#ACS: 坐标系统号错误 '

检测时间： 编辑程序时。

引起原因： #CS DEF、#CS ON、#CS NEW、#ACS DEF、#ACS ON 或 #ACS NEW 指令中的坐标系统号编程值错误。

解决方案： 系统号的值必须介于 1 ～ 5。

8314 'G30/G73 编程错误 '

检测时间： 编辑程序时。

引起原因： G30/G73 功能中只编写了参数 I 或 J 中的一个。

解决方案： 要么同时编写参数 I 和 J，要么都不编写。

8315 ' 探测循环号错误 '

检测时间： 在 MDI 模式中编辑和执行程序时。

引起原因： 编写的探测循环号错误。

解决方案： 正确的探测循环号为 PROBE1 ～ PROBE8。

8316 ' 参数不允许使用 '

检测时间： 在 MDI 模式中编辑和执行程序时。

引起原因： 编写的指令参数错误。

解决方案： 参考编程手册，确保每个循环和指令的参数正确。

8317 'D 功能为负 '

检测时间： 编辑程序时。

引起原因： 编写了负的刀具偏置。

解决方案： 刀具偏置值必须为正。

8318 ' 平均每个程序段只允许使用一根活动轴 1'

检测时间： 编辑程序时。

引起原因： 程序段中重复编写 @1 轴。

解决方案： 只在程序段中编写一次 @1。



8319 ' 平均每个程序段只允许使用一根活动轴 2'

检测时间： 编辑程序时。
引起原因： 程序段中重复编写 @2 轴。
解决方案： 只在程序段中编写一次 @2。

8320 ' 平均每个程序段只允许使用一根活动轴 3'

检测时间： 编辑程序时。
引起原因： 程序段中重复编写 @3 轴。
解决方案： 只在程序段中编写一次 @3。

8321 ' 平均每个程序段只允许使用一根活动轴 4 '

检测时间： 编辑程序时。
引起原因： 程序段中重复编写 @4 轴。
解决方案： 只在程序段中编写一次 @4。

8322 ' 平均每个程序段只允许使用一根活动轴 5'

检测时间： 编辑程序时。
引起原因： 程序段中重复编写 @5 轴。
解决方案： 只在程序段中编写一次 @5。

8323 ' 需要编写等号 '

检测时间： 编辑程序时。
引起原因： "=" 替代 "==" 使用。
解决方案： 编写 "=="，而不是 "="。

8327 ' 某些圆括号缺失 '

检测时间： 在 MDI 模式中编辑和执行程序时。
引起原因： 左圆括号的数量与右圆括号的数量不匹配。
解决方案： 检查每个左圆括号是否有对应的右圆括号。

8328 '#SET AX: 只允许使用 0 位置 '

检测时间： 编辑程序时。
引起原因： #SET AX 指令中编写了整数值，而不是 0。
解决方案： #SET AX 指令只能使用 0 值来表示该位置未被任何轴占用。

8329 ' 需要 V'

检测时间： 编辑程序时。
引起原因： 编写的变量名不是以 "V." 开头。
解决方案： 工件程序和通过 MDI 定义的变量必须以前缀 "V." 开始。

8330 ' 需要轴名或轴号 '

检测时间： 在 MDI 模式中编辑和执行程序时。
引起原因： 编程指令或变量需要轴名或轴号。
解决方案： 查阅编程手册，检查指令或变量的语法。

8331 ' 不需要轴名或轴号 '

检测时间： 在 MDI 模式中编辑和执行程序时。
引起原因： 编程指令或变量不允许使用轴名或轴号。
解决方案： 查阅编程手册，检查指令或变量的语法。

8402 ' 刀库号错误 '

检测时间： 编辑程序时。
引起原因： 刀具管理器变量请求的刀库号错误。
解决方案： 刀库号必须介于 1 ～ 4 之间。如果未指定刀库号，则默认为第一个。





刀具表与刀库表

‘ 刀具不能加载到位置 %1’

检测时间： 从对话框或刀具清单中手动加载刀具或加载刀库表时。

引起原因： 刀具不能加载到该位置上，可能是该刀具已经位于刀库中，或者未在刀具表中对其进行定义。

解决方案： 选择足够大的刀库位置。确保已经在刀具表中定义了刀具。

‘ 不能卸载 %1 刀具 ’

检测时间： 从对话框或刀库清单中手动卸载刀具或加载刀具表或刀库表时。

引起原因： 刀具不在刀库中。

解决方案： 选择位于刀库中的刀具。

‘ 不能执行机动，确保主轴为空 ’

检测时间： 执行基础刀具加载或卸载坯料时。

引起原因： 主轴不为空或某通道处于错误状态。

解决方案： 从主轴卸载刀具。确保所有通道的状态为 “READY（准备完毕）”。

‘ 选择的表类型错误 ’

检测时间： 加载刀具表或刀库表时。

引起原因： 被加载文件与表不对应或在外部完成编辑。

解决方案： 选择与待加载表对应的文件。

' 刀具不存在或已经加载到刀库中 '

检测时间： 机动或非机动加载刀具至刀库时。

引起原因： 刀具不存在或已经位于刀库中，位于主轴上或换刀夹臂中。

解决方案： 定义一把刀具。如果该刀具位于主轴或换刀夹臂中，将其加载到刀库中。

‘ 不能删除刀具 %1’

检测时间： 删除刀具或加载刀具表时。

引起原因： 不能找到该刀。

解决方案： 刀具缺失，因而不能删除。如果在加载刀具表时出现该问题，使用软键初始化该表，然后重新加载。

' 刀具 %1 不存在或未加载到刀库中 '

检测时间： 从刀库中卸载刀具时，抢占刀具位置时或加载刀库表时。

引起原因： 刀具不存在或未加载到刀库中。

解决方案： 定义刀具并将其加载到刀库中。



‘ 重命名刀具错误 ’

检测时间： 更改刀具名时。

引起原因： 不能在数据库中创建刀具 (bd8070. mdb)。数据库正在被另外的应用程序使用，或者它没有写操作权限。

解决方案： 数据库必须拥有写操作权限。如果刀具数据库正被其他应用程序使用，等待，直至它被释放。

‘ 不能重命名位于主轴、刀库或换刀夹臂中的刀具 ’

检测时间： 更改刀具名时。

引起原因： 试图更改刀具名，但是相同名字的刀具已经位于主轴、刀库或换刀夹臂中。

解决方案： 选择其他编号，或者卸载刀具。

‘ 如果换刀夹臂 2 上有刀具，则不能在主轴上放置刀具 ’

检测时间： 将刀具强制设置为主轴刀具时。

引起原因： 换刀夹臂 2 上已经有一把刀。

解决方案： 从换刀夹臂 2 上移除刀具。

‘ 不能将换刀夹臂上的刀具置于主轴上 ’

检测时间： 将刀具强制设置为主轴刀具时。

引起原因： 刀具位于换刀夹臂 1 上。

解决方案： 从换刀夹臂 1 上移除刀具。

‘ 位置不存在 ’

检测时间： 机动或非机动加载或从刀库卸载刀具时。

引起原因： 刀库位置不存在。

解决方案： 选择已有的刀库位置或创建该位置 (机床参数)。

‘ 未找到路径 %1’

检测时间： 加载、保存或打印刀具表或刀库表时。

引起原因： 选择的路径不存在。

解决方案： 正确定义路径或创建该路径。

‘ 未找到文件 %1’

检测时间： 加载刀具表或刀库表时。

引起原因： 数据文件不存在。

解决方案： 选择已有的文件并正确指定其路径。

‘ 将数据保存到 %1 出错 ’

检测时间： 保存刀具表或刀库表时。

引起原因： 不能创建数据文件、该文件已经存在且无写操作权限、该文件正被使用或者磁盘空间不够。

解决方案： 选择其他目录保存表格、赋予写操作权限、关闭正在使用该文件的应用程序或者释放磁盘空间。

' 关闭文件 %1 错误 '

检测时间： 加载、保存或打印刀具表或刀库表时。

引起原因： 可能的原因包括：

1. 加载表格时。数据文件已经存在、没有读操作权限或正被另一应用程序使用 。
2. 保存表格时。不能创建数据文件、该文件已经存在且无写操作权限、该文件正被使用或者磁盘空间不够。
3. 打印表格时。打印到文件时，不能创建数据文件、该文件已经存在且无写操作权限、该文件正被使用或者磁盘空间不够。打印到打印机时，打印机不存在或配置错误。

解决方案： 可采用的解决办法有：

1. 加载表格时。选择已有的文件、正确指定其路径、赋予读操作权限或关闭正在使用该表格的应用程序。
2. 保存表格时。选择其他目录保存该表格、赋予写操作权限、关闭正在使用该表格的应用程序或释放磁盘空间。
3. 打印表格时。打印到文件时，选择其他目录保存该表格、赋予写操作权限、关闭正在使用该表格的应用程序或者释放磁盘空间。打印到打印机时，选择现有的配置正确的打印机。



轮廓编辑器信息

‘剖面未解决’

引起原因： 在 “FINISHING（精加工）” 或 “SAVING AND CONTINUING（保存后继续）” 时，待保存的剖面尚未完全加工完。

解决方案： 解决待保存的剖面。只能保存解决完的剖面。

‘内存不够’

引起原因： 系统内存不够，剖面编辑器无法继续工作。

‘几何尺寸错误’

引起原因： 利用所选的剖面访问剖面编辑器时出现几何尺寸错误。所选剖面的路径定义中存在某些错误数据。

解决方案： 纠正剖面。正确定义所有定义剖面的路径。

‘元素数据错误’

引起原因： 输入元素的数据错误。

解决方案： 纠正元素数据。

‘剖面数据出错’

引起原因： 编辑 “圆形” 或 “矩形” 剖面的数据错误。

解决方案： 纠正元素数据。

‘圆弧未经过其起点坐标’

引起原因： 圆弧的圆心、半径以及起点坐标数据不一致。

解决方案： 纠正元素数据。

‘圆弧未经过其终点坐标’

引起原因： 圆弧的圆心、半径以及终点坐标数据不一致。

解决方案： 纠正元素数据。

‘没有圆弧能与所有数据相符’

引起原因： 不能找到与已知数据相符的圆弧。

解决方案： 纠正元素数据。

‘元素不能与前一元素相切’

引起原因： 输入与前一元素相切的元素中的数据不连贯。

解决方案： 纠正元素数据。

'值错误'

引起原因： 输入的用以修改拐点 (圆角、倒角、切入或切出) 的值错误。

解决方案： 纠正元素数据。拐点的值必须小于它的定义路径。

‘平面的轴错误’

引起原因： 平面的某根轴错误。某些轴位于平面的 “CONFIGURATION” 里，或者 CNC 未定义所选剖面的轴。

解决方案： 必须通过两根轴形成一个平面。两根轴必须位于 CNC 中。




FAGOR






FAGOR












FAGOR












FAGOR



